Bao-Liang Lu
Liqing Zhang
James Kwok (Eds.)

Neural Information Processing

18th International Conference, ICONIP 2011
Shanghai, China, November 2011
Proceedings, Part III

3 Part III

Springer

Lecture Notes in Computer Science 7064

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Bao-Liang Lu   Liqing Zhang
James Kwok (Eds.)

Neural
Information Processing

18th International Conference, ICONIP 2011
Shanghai, China, November 13-17, 2011
Proceedings, Part III

 Springer

Volume Editors

Bao-Liang Lu
Shanghai Jiao Tong University
Department of Computer Science and Engineering
800, Dongchuan Road, Shanghai 200240, China
E-mail: bllu@sjtu.edu.cn

Liqing Zhang
Shanghai Jiao Tong University
Department of Computer Science and Engineering
800, Dongchuan Road, Shanghai 200240, China
E-mail: zhang-lq@cs.sjtu.edu.cn

James Kwok
The Hong Kong University of Science and Technology
Department of Computer Science and Engineering
Clear Water Bay, Kowloon, Hong Kong, China
E-mail: jamesk@cse.ust.hk

Preface

This book and its sister volumes constitute the proceedings of the 18th International Conference on Neural Information Processing (ICONIP 2011) held in Shanghai, China, during November 13–17, 2011. ICONIP is the annual conference of the Asia Pacific Neural Network Assembly (APNNA). ICONIP aims to provide a high-level international forum for scientists, engineers, educators, and students to address new challenges, share solutions, and discuss future research directions in neural information processing and real-world applications.

The scientific program of ICONIP 2011 presented an outstanding spectrum of over 260 research papers from 42 countries and regions, emerging from multidisciplinary areas such as computational neuroscience, cognitive science, computer science, neural engineering, computer vision, machine learning, pattern recognition, natural language processing, and many more to focus on the challenges of developing future technologies for neural information processing. In addition to the contributed papers, we were particularly pleased to have 10 plenary speeches by world-renowned scholars: Shun-ichi Amari, Kunihiko Fukushima, Aike Guo, Lei Xu, Jun Wang, DeLiang Wang, Derong Liu, Xin Yao, Soo-Young Lee, and Nikola Kasabov. The program also includes six excellent tutorials by David Cai, Irwin King, Pei-Ji Liang, Hiroshi Mamitsuka, Ming Zhou, Hang Li, and Shanfeng Zhu. The conference was followed by three post-conference workshops held in Hangzhou, on November 18, 2011: "ICONIP2011Workshop on Brain – Computer Interface and Applications," organized by Bao-Liang Lu, Liqing Zhang, and Chin-Teng Lin; "The 4th International Workshop on Data Mining and Cybersecurity," organized by Paul S. Pang, Tao Ban, Youki Kadobayashi, and Jungsuk Song; and "ICONIP 2011 Workshop on Recent Advances in Nature-Inspired Computation and Its Applications," organized by Xin Yao and Shan He.

The ICONIP 2011 organizers would like to thank all special session organizers for their effort and time high enriched the topics and program of the conference. The program included the following 13 special sessions: "Advances in Computational Intelligence Methods-Based Pattern Recognition," organized by Kai-Zhu Huang and Jun Sun; "Biologically Inspired Vision and Recognition," organized by Jun Miao, Libo Ma, Liming Zhang, Juyang Weng and Xilin Chen; "Biomedical Data Analysis," organized by Jie Yang and Guo-Zheng Li; "Brain Signal Processing," organized by Jian-Ting Cao, Tomasz M. Rutkowski, Toshihisa Tanaka, and Liqing Zhang; "Brain-Realistic Models for Learning, Memory and Embodied Cognition," organized by Huajin Tang and Jun Tani; "Clifford Algebraic Neural Networks," organized by Tohru Nitta and Yasuaki Kuroe; "Combining Multiple Learners," organized by Younès Bennani, Nistor Grozavu, Mohamed Nadif, and Nicoleta Rogovschi; "Computational Advances in Bioinformatics," organized by Jonathan H. Chan; "Computational-Intelligent Human–Computer Interaction," organized by Chin-Teng Lin, Jyh-Yeong Chang,

John Kar-Kin Zao, Yong-Sheng Chen, and Li-Wei Ko; "Evolutionary Design and Optimization," organized by Ruhul Sarker and Mao-Lin Tang; "Human-Originated Data Analysis and Implementation," organized by Hyeyoung Park and Sang-Woo Ban; "Natural Language Processing and Intelligent Web Information Processing," organized by Xiao-Long Wang, Rui-Feng Xu, and Hai Zhao; and "Integrating Multiple Nature-Inspired Approaches," organized by Shan He and Xin Yao.

The ICONIP 2011 conference and post-conference workshops would not have achieved their success without the generous contributions of many organizations and volunteers. The organizers would also like to express sincere thanks to APNNA for the sponsorship, to the China Neural Networks Council, International Neural Network Society, and Japanese Neural Network Society for their technical co-sponsorship, to Shanghai Jiao Tong University for its financial and logistic supports, and to the National Natural Science Foundation of China, Shanghai Hyron Software Co., Ltd., Microsoft Research Asia, Hitachi (China) Research & Development Corporation, and Fujitsu Research and Development Center, Co., Ltd. for their financial support.

We are very pleased to acknowledge the support of the conference Advisory Committee, the APNNA Governing Board and Past Presidents for their guidance, and the members of the International Program Committee and additional reviewers for reviewing the papers. Particularly, the organizers would like to thank the proceedings publisher, Springer, for publishing the proceedings in the *Lecture Notes in Computer Science Series*. We want to give special thanks to the Web managers, Haoyu Cai and Dong Li, and the publication team comprising Li-Chen Shi, Yong Peng, Cong Hui, Bing Li, Dan Nie, Ren-Jie Liu, Tian-Xiang Wu, Xue-Zhe Ma, Shao-Hua Yang, Yuan-Jian Zhou and Cong Xie for checking the accepted papers in a short period of time. Last but not least, the organizers would like to thank all the authors, speakers, audience, and volunteers.

November 2011   Bao-Liang Lu
                                                                  Liqing Zhang
                                                                  James Kwok

ICONIP 2011 Organization

Organizer

Shanghai Jiao Tong University

Sponsor

Asia Pacific Neural Network Assembly

Financial Co-sponsors

Shanghai Jiao Tong University
National Natural Science Foundation of China
Shanghai Hyron Software Co., Ltd.
Microsoft Research Asia
Hitachi (China) Research & Development Corporation
Fujitsu Research and Development Center, Co., Ltd.

Technical Co-sponsors

China Neural Networks Council
International Neural Network Society
Japanese Neural Network Society

Honorary Chair

Shun-ichi Amari   Brain Science Institute, RIKEN, Japan

Advisory Committee Chairs

Shoujue Wang   Institute of Semiconductors,
                                     Chinese Academy of Sciences, China
Aike Guo   Institute of Neuroscience, Chinese Academy of
                                     Sciences, China
Liming Zhang   Fudan University, China

Advisory Committee Members

Sabri Arik	Istanbul University, Turkey
Jonathan H. Chan	King Mongkut's University of Technology, Thailand
Wlodzislaw Duch	Nicolaus Copernicus University, Poland
Tom Gedeon	Australian National University, Australia
Yuzo Hirai	University of Tsukuba, Japan
Ting-Wen Huang	Texas A&M University, Qatar
Akira Hirose	University of Tokyo, Japan
Nik Kasabov	Auckland University of Technology, New Zealand
Irwin King	The Chinese University of Hong Kong, Hong Kong
Weng-Kin Lai	MIMOS, Malaysia
Min-Ho Lee	Kyungpoor National University, Korea
Soo-Young Lee	Korea Advanced Institute of Science and Technology, Korea
Andrew Chi-Sing Leung	City University of Hong Kong, Hong Kong
Chin-Teng Lin	National Chiao Tung University, Taiwan
Derong Liu	University of Illinois at Chicago, USA
Noboru Ohnishi	Nagoya University, Japan
Nikhil R. Pal	Indian Statistical Institute, India
John Sum	National Chung Hsing University, Taiwan
DeLiang Wang	Ohio State University, USA
Jun Wang	The Chinese University of Hong Kong, Hong Kong
Kevin Wong	Murdoch University, Australia
Lipo Wang	Nanyang Technological University, Singapore
Xin Yao	University of Birmingham, UK
Liqing Zhang	Shanghai Jiao Tong University, China

General Chair

Bao-Liang Lu	Shanghai Jiao Tong University, China

Program Chairs

Liqing Zhang	Shanghai Jiao Tong University, China
James T.Y. Kwok	Hong Kong University of Science and Technology, Hong Kong

Organizing Chair

Hongtao Lu	Shanghai Jiao Tong University, China

Workshop Chairs

Guangbin Huang   Nanyang Technological University, Singapore
Jie Yang   Shanghai Jiao Tong University, China
Xiaorong Gao   Tsinghua University, China

Special Sessions Chairs

Changshui Zhang   Tsinghua University, China
Akira Hirose   University of Tokyo, Japan
Minho Lee   Kyungpoor National University, Korea

Tutorials Chair

Si Wu   Institute of Neuroscience, Chinese Academy of
                        Sciences, China

Publications Chairs

Yuan Luo   Shanghai Jiao Tong University, China
Tianfang Yao   Shanghai Jiao Tong University, China
Yun Li   Nanjing University of Posts and
                        Telecommunications, China

Publicity Chairs

Kazushi Ikeda   Nara Institute of Science and Technology,
                        Japan
Shaoning Pang   Unitec Institute of Technology, New Zealand
Chi-Sing Leung   City University of Hong Kong, China

Registration Chair

Hai Zhao   Shanghai Jiao Tong University, China

Financial Chair

Yang Yang   Shanghai Maritime University, China

Local Arrangements Chairs

Guang Li   Zhejiang University, China
Fang Li   Shanghai Jiao Tong University, China

Secretary

Xun Liu   Shanghai Jiao Tong University, China

Program Committee

Shigeo Abe	Takio Kurita
Bruno Apolloni	Minho Lee
Sabri Arik	Chi Sing Leung
Sang-Woo Ban	Chunshien Li
Jianting Cao	Guo-Zheng Li
Jonathan Chan	Junhua Li
Songcan Chen	Wujun Li
Xilin Chen	Yuanqing Li
Yen-Wei Chen	Yun Li
Yiqiang Chen	Huicheng Lian
Siu-Yeung David Cho	Peiji Liang
Sung-Bae Cho	Chin-Teng Lin
Seungjin Choi	Hsuan-Tien Lin
Andrzej Cichocki	Hongtao Lu
Jose Alfredo Ferreira Costa	Libo Ma
Sergio Cruces	Malik Magdon-Ismail
Ke-Lin Du	Robert(Bob) McKay
Simone Fiori	Duoqian Miao
John Qiang Gan	Jun Miao
Junbin Gao	Vinh Nguyen
Xiaorong Gao	Tohru Nitta
Nistor Grozavu	Toshiaki Omori
Ping Guo	Hassab Elgawi Osman
Qing-Long Han	Seiichi Ozawa
Shan He	Paul Pang
Akira Hirose	Hyeyoung Park
Jinglu Hu	Alain Rakotomamonjy
Guang-Bin Huang	Sarker Ruhul
Kaizhu Huang	Naoyuki Sato
Amir Hussain	Lichen Shi
Danchi Jiang	Jochen J. Steil
Tianzi Jiang	John Sum
Tani Jun	Jun Sun
Joarder Kamruzzaman	Toshihisa Tanaka
Shunshoku Kanae	Huajin Tang
Okyay Kaynak	Maolin Tang
John Keane	Dacheng Tao
Sungshin Kim	Qing Tao
Li-Wei Ko	Peter Tino

Ivor Tsang
Michel Verleysen
Bin Wang
Rubin Wang
Xiao-Long Wang
Yimin Wen
Young-Gul Won
Yao Xin
Rui-Feng Xu
Haixuan Yang
Jie Yang

Yang Yang
Yingjie Yang
Zhirong Yang
Dit-Yan Yeung
Jian Yu
Zhigang Zeng
Jie Zhang
Kun Zhang
Hai Zhao
Zhihua Zhou

Reviewers

Pablo Aguilera
Lifeng Ai
Elliot Anshelevich
Bruno Apolloni
Sansanee
    Auephanwiriyakul
Hongliang Bai
Rakesh Kr Bajaj
Tao Ban
Gang Bao
Simone Bassis
Anna Belardinelli
Yoshua Bengio
Sergei Bezobrazov
Yinzhou Bi
Alberto Borghese
Tony Brabazon
Guenael Cabanes
Faicel Chamroukhi
Feng-Tse Chan
Hong Chang
Liang Chang
Aaron Chen
Caikou Chen
Huangqiong Chen
Huanhuan Chen
Kejia Chen
Lei Chen
Qingcai Chen
Yin-Ju Chen

Yuepeng Chen
Jian Cheng
Wei-Chen Cheng
Yu Cheng
Seong-Pyo Cheon
Minkook Cho
Heeyoul Choi
Yong-Sun Choi
Shihchieh Chou
Angelo Ciaramella
Sanmay Das
Satchidananda Dehuri
Ivan Duran Diaz
Tom Diethe
Ke Ding
Lijuan Duan
Chunjiang Duanmu
Sergio Escalera
Aiming Feng
Remi Flamary
Gustavo Fontoura
Zhenyong Fu
Zhouyu Fu
Xiaohua Ge
Alexander Gepperth
M. Mohamad Ghassany
Adilson Gonzaga
Alexandre Gravier
Jianfeng Gu
Lei Gu

Zhong-Lei Gu
Naiyang Guan
Pedro Antonio Gutiérrez
Jing-Yu Han
Xianhua Han
Ross Hayward
Hanlin He
Akinori Hidaka
Hiroshi Higashi
Arie Hiroaki
Eckhard Hitzer
Gray Ho
Kevin Ho
Xia Hua
Mao Lin Huang
Qinghua Huang
Sheng-Jun Huang
Tan Ah Hwee
Kim Min Hyeok
Teijiro Isokawa
Wei Ji
Zheng Ji
Caiyan Jia
Nanlin Jin
Liping Jing
Yoonseop Kang
Chul Su Kim
Kyung-Joong Kim
Saehoon Kim
Yong-Deok Kim

Irwin King
Jun Kitazono
Masaki Kobayashi
Yasuaki Kuroe
Hiroaki Kurokawa
Chee Keong Kwoh
James Kwok
Lazhar Labiod
Darong Lai
Yuan Lan
Kittichai
     Lavangnananda
John Lee
Maylor Leung
Peter Lewis
Fuxin Li
Gang Li
Hualiang Li
Jie Li
Ming Li
Sujian Li
Xiaosong Li
Yu-feng Li
Yujian Li
Sheng-Fu Liang
Shu-Hsien Liao
Chee Peng Lim
Bingquan Liu
Caihui Liu
Jun Liu
Xuying Liu
Zhiyong Liu
Hung-Yi Lo
Huma Lodhi
Gabriele Lombardi
Qiang Lu
Cuiju Luan
Abdelouahid Lyhyaoui
Bingpeng Ma
Zhiguo Ma
Laurens Van Der Maaten
Singo Mabu
Shue-Kwan Mak
Asawin Meechai
Limin Meng

Komatsu Misako
Alberto Moraglio
Morten Morup
Mohamed Nadif
Kenji Nagata
Quang Long Ngo
Phuong Nguyen
Dan Nie
Kenji Nishida
Chakarida Nukoolkit
Robert Oates
Takehiko Ogawa
Zeynep Orman
Jonathan
     Ortigosa-Hernandez
Mourad Oussalah
Takashi J. Ozaki
Neyir Ozcan
Pan Pan
Paul S. Pang
Shaoning Pang
Seong-Bae Park
Sunho Park
Sakrapee Paul
Helton Maia Peixoto
Yong Peng
Jonas Peters
Somnuk
     Phon-Amnuaisuk
J.A. Fernandez Del Pozo
Santitham Prom-on
Lishan Qiao
Yuanhua Qiao
Laiyun Qing
Yihong Qiu
Shah Atiqur Rahman
Alain Rakotomamonjy
Leon Reznik
Nicoleta Rogovschi
Alfonso E. Romero
Fabrice Rossi
Gain Paolo Rossi
Alessandro Rozza
Tomasz Rutkowski
Nishimoto Ryunosuke

Murat Saglam
Treenut Saithong
Chunwei Seah
Lei Shi
Katsunari Shibata
A. Soltoggio
Bo Song
Guozhi Song
Lei Song
Ong Yew Soon
Liang Sun
Yoshinori Takei
Xiaoyang Tan
Chaoying Tang
Lei Tang
Le-Tian Tao
Jon Timmis
Yohei Tomita
Ming-Feng Tsai
George Tsatsaronis
Grigorios Tsoumakas
Thomas Villmann
Deng Wang
Frank Wang
Jia Wang
Jing Wang
Jinlong Wang
Lei Wang
Lu Wang
Ronglong Wang
Shitong Wang
Shuo Wang
Weihua Wang
Weiqiang Wang
Xiaohua Wang
Xiaolin Wang
Yuanlong Wang
Yunyun Wang
Zhikun Wang
Yoshikazu Washizawa
Bi Wei
Kong Wei
Yodchanan Wongsawat
Ailong Wu
Jiagao Wu

Jianxin Wu
Qiang Wu
Si Wu
Wei Wu
Wen Wu
Bin Xia
Chen Xie
Zhihua Xiong
Bingxin Xu
Weizhi Xu
Yang Xu
Xiaobing Xue
Dong Yang
Wei Yang
Wenjie Yang
Zi-Jiang Yang
Tianfang Yao
Nguwi Yok Yen
Florian Yger
Chen Yiming
Jie Yin
Lijun Yin
Xucheng Yin
Xuesong Yin

Jiho Yoo
Washizawa Yoshikazu
Motohide Yoshimura
Hongbin Yu
Qiao Yu
Weiwei Yu
Ying Yu
Jeong-Min Yun
Zeratul Mohd Yusoh
Yiteng Zhai
Biaobiao Zhang
Danke Zhang
Dawei Zhang
Junping Zhang
Kai Zhang
Lei Zhang
Liming Zhang
Liqing Zhang
Lumin Zhang
Puming Zhang
Qing Zhang
Rui Zhang
Tao Zhang
Tengfei Zhang

Wenhao Zhang
Xianming Zhang
Yu Zhang
Zehua Zhang
Zhifei Zhang
Jiayuan Zhao
Liang Zhao
Qi Zhao
Qibin Zhao
Xu Zhao
Haitao Zheng
Guoqiang Zhong
Wenliang Zhong
Dong-Zhuo Zhou
Guoxu Zhou
Hongming Zhou
Rong Zhou
Tianyi Zhou
Xiuling Zhou
Wenjun Zhu
Zhanxing Zhu
Fernando José Von Zube

Table of Contents – Part III

Multi-agent Systems

Natural Language Processing and Intelligent Web Information Processing

Neural Encoding and Decoding

Neural Network Models

Neuromorphic Hardware and Implementations

Object Recognition

Visual Perception Modelling

Advances in Computational Intelligence Methods Based Pattern Recognition

Multimodal Identity Verification
Based on Learning Face and Gait Cues

Emdad Hossain and Girija Chetty

Faculty of Information Sciences and Engineering, University of Canberra, Australia
emdad.hossain@canberra.edu.au

Abstract. In this paper we propose a novel multimodal Bayesian approach based on PCA-LDA processing for person identification from low resolution surveillance video with cues extracted from gait and face biometrics. The experimental evaluation of the proposed scheme on a publicly available database [2] showed that the combined PCA-LDA face and gait features can lead to powerful identity verification and can capture the inherent multimodality in walking gait patterns and discriminate the identity from low resolution surveillance videos.

Keywords: Biometric, gait recognition, PCA, LDA, Bayesian, k-NN.

1   Introduction

Human identity verification from arbitrary views is a very challenging problem, especially when one is walking at a distance. Lately, recognizing identity from gait patterns has become a popular area of research in biometrics and computer vision, and one of the most successful applications of image analysis and understanding. Gait recognition is one of new and important biometric technologies based on behavioral characteristics, and it involves identifying individuals by their walking patterns. Gait can be captured at a distance by using low resolution devices, while other biometrics needs higher resolution. Gait is difficult to disguise, and can be performed at a distance or at low resolution and requires no body-invading equipment to capture gait information. Gait recognition can hence be considered as a powerful recognition technology for next-generation surveillance and access control applications, with applicability to many civilian and high security environments such as airports, banks, military bases, car parks, railway stations etc. Further, gait is an inherently multimodal biometric as proposed in [1], suggesting that there are 24 different components to human gait, and involves not only the lower body but also the upper body motion, including head and the hands. If all gait movements from full body images can be captured, it can be a truly an unique biometric for ascertaining identity. In this paper we propose a novel approach based on learning face and gait features in transform subspaces. And show even without inclusion of dynamic gait features, it is possible to obtain a significant improvement in recognition performance, provided appropriate transform subspaces are considered. We examined two such multivariate statistical subspaces based on principal component analysis (PCA) and linear

discriminant analysis (LDA and fusion of face and gait features. Extensive experiments conducted on a publicly available gait database [2] suggest that the proposed approach can capture several inherent multimodal components from gait, and face of a walking human from low resolution video. Even without dynamic cues, a simple, practical and robust identity verification system can be built in spite of poor quality data from surveillance video, and significant pose and illumination variations.

Rest of the paper is organized as follows. Next Section discusses the background and the previous work, followed by our proposed scheme in Section 3. In Section 4 we describe the details of the experimental work carried out, and a discussion on some of the results obtained from the experimental work. The paper concludes in Section 6 with conclusions and plan for further work.

2 Background

Current state-of-the-art video surveillance systems, when used for recognizing the identity of the person in the scene, cannot perform very well due to low quality video or inappropriate processing techniques. Though much progress has been made in the past decade on visual based automatic person identification through utilizing different biometrics, including face recognition, gait analysis, iris and fingerprint recognition, each of these techniques work satisfactorily in highly controlled operating environments such as border control or immigration check points, under constrained illumination, pose and facial expressions. To address the next generation security and surveillance requirements and for diffusion of biometrics based security systems for day-to-day civilian access control applications, we need a robust and invariant biometric trait [3] to identify a person for both controlled and uncontrolled operational environments.

Face recognition has been the focus of extensive research for the past three decades [3]. The approaches for this task can be broadly divided into two categories: 1) Feature-based methods [4, 5], which first process the input image to identify and extract distinctive facial features such as the eyes, mouth, nose, etc. as well as other fiducially marks and then compute the geometric relationships among those facial points, thus, reducing the input facial image to a vector of geometric features. Standard statistical pattern recognition techniques are then employed for matching faces using these measurements. 2) Appearance-based (or holistic) methods [6, 7], which attempt to identify faces using global representations, i.e., descriptions based on the entire image rather than on local features of the face. Though face recognition methods traditionally operate on static intensity images. In recent years, much effort has also been directed towards identifying faces from video [8] as well as from other modalities such as 3D [9] and infra-red [10].

Recently, much effort has been expended on combining various biometrics in a bid to improve upon the recognition accuracy of classifiers that are based on a single biometric. Some biometric combinations which have been tested include face, fingerprint and hand geometry [11]; face, fingerprint and speech [12]; face and iris [13]; face and ear [14]; and face and speech [15, 16, 17]. The potential of gait as a powerful biometric has been explored in some of the recent works [18, 19], though inherent multimodal components present in the whole body during walking has not

been much exploited by the research community. In this paper we explore some preliminary work on how these multimodal aspects can play an important role in differentiating individuals during walking.

On another note, some of the most important challenges for diffusion of biometrics in day-to-day civilian applications are issues related to invasion of privacy. In [20], an extensive study has shown that physiological biometrics as having no negative impact on privacy. That is an excellent motivation for us to investigate face, body and gait cues during walking as a powerful biometric with inherent multimodality for establishing the identity of a person. Further, these video based cues can be captured remotely from a distance, and by using an appropriate biometric identification protocol such as the one suggested by authors in [21], it can be ensured that sensitive privacy concerns are addressed as well. An appropriate protocol as in [21] can ensure that the identification system is not misused and that function creep (i.e. use for another purpose is prevented). This means in particular that a component should not be able to learn more information than what is really needed for a correct result. In fact our proposed fusion of side face, body and gait cues captured from low resolution surveillance videos ("security check: pass") needs strong algorithms and processing techniques to be of any use for establishing identity, and of no use without them, and safe-guard the privacy to some extent automatically. The details of the publicly available gait database used for this research, and the proposed multimodal identification scheme are described in the next Section.

3   Multimodal Identification Scheme

For experimental evaluation of our proposed multimodal scheme, we used a publicly available video database of human actions [2]. This video database contains six types of human actions (walking, jogging, running, boxing, hand waving and hand clapping) performed several times by 25 subjects in four different scenarios: outdoors s1, outdoors with scale variation s2, outdoors with different clothes s3 and indoors s4. Currently the database contains 2391 sequences.

Fig. 1. Sample images from human action database for walking sequences [2]

All sequences were taken over homogeneous backgrounds with a static camera with 25fps frame rate. The sequences were down-sampled to the spatial resolution of 160×120 pixels and have a length of four seconds in average. We used only the walking sequences for our experiments and Figure 1 shows some of the sample images from the walking video sequences.

For all our experiments we used 100 video sequences for 25 people. There were 19 males and 6 females in the entire walking dataset. We performed some image pre-processing steps corresponding to background segmentation, cropping, filtering and histogram equalization of images of the walking human and then extracted features based on PCA (principal component analysis) and LDA (linear discriminant analysis). We used separate set for performing training and testing. The low dimensional PCA and LDA features were then classified by a Bayesian classifier.
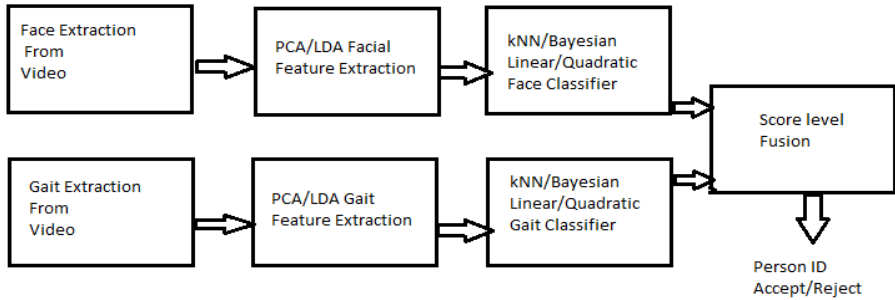
Fig. 2(a). Score-level Fusion for proposed multimodal identification scheme based on side face and gait cues extracted from low-resolution video
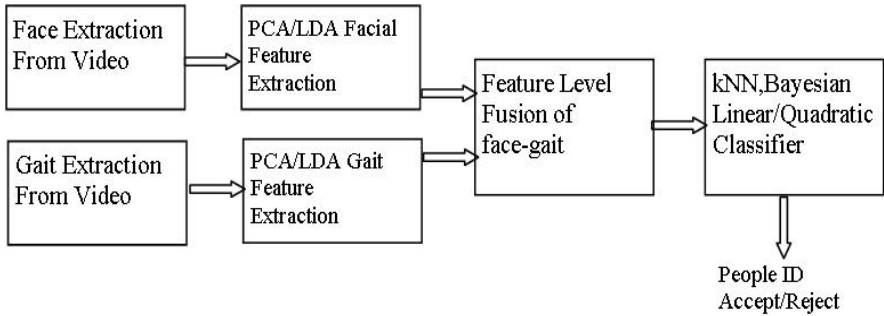
Fig. 2(b). Feature-level fusions for proposed multimodal identification scheme based on side face and gait cues extracted from low-resolution video

We examined three different classifiers, the nearest neighbor (*k-NN*), the Bayesian linear and the Bayesian quadratic classifiers. The combination of the low dimensional, discriminative PCA and LDA features along with powerful Bayesian classifiers allow us to achieve significant improvement in recognition accuracy. This is because Bayesian classifiers have the flexibility to incorporate prior information, and can predict how a system's performance will change when going from one environment to another or when going from one type of testing to another [23]. And *k-NN* is very effective simple classifier with noise reduction capabilities [24]. Further, we examined two different fusion methods – feature-level fusion and score-level fusion

of face and gait features. The schematic for the proposed multimodal identification scheme is shown in Figure 2(a) and 2(b).

From the segmented face and gait gray-scale images, we extract the PCA and LDA features. As more than 95% variations were captured in first 40 dimensions, the length of face and gait PCA and LDA vectors were kept at 40. For score-level fusion, we used SUM rule with equal weights for face and gait features. For feature-level fusion, we concatenated face and gait features into a one long vector.

The Bayesian linear and quadratic discriminant classifier uses Bayesian decision rule for classifying a set of learned feature vectors to a class [27]. While the linear classifier fits a multivariate normal density to each group, with a pooled estimate of covariance, the quadratic discriminant classifier fits MVN (multivariate normal) densities with covariance estimates stratified by group. Both methods use likelihood ratios to assign observations to groups. Given a set of classes M characterized by a set of known parameters in model Ω a set of extracted feature vector X belongs to the class which has the highest probability. This is shown in Eq.(1)) and is known as Bayesian decision rule.

$$ X \in M_k \qquad P\big(M_k\big|X,\Omega\big) \geq P\big(M_l\big|X,\Omega\big) \qquad \forall l \neq k \ \dots \qquad \dots\dots\dots(1) $$

To calculate the a-posteriori probability shown, we used Bayesian law of statistics which finally by assuming that features are distributed normally, leads to a quadratic classifier format known as Bayesian Quadratic classifier [26]. The model Ω consists of the mean and the covariance of our training vectors, and likelihoods are calculated as stated above. The details of the experiments carried out is discussed in the next Section.

4   Experimental Results and Discussion

We performed different sets of experiments for examining the discriminating ability of proposed feature extraction transformation and classifier techniques. Further we also compared the performance of score and feature-level fusion (schematic shown in Figure 2(a) and 2(b)) The recognition performance of single mode face and gait features, and with fusion of face and gait features at score-level and at feature-level, are discussed in next few sub-sections.

4.1   Recognition Performance with PCA-Features

For the first set of experiments we used PCA features for training and testing, with a Bayesian (linear/quadratic) and k-nearest neighbor classifiers for classification. Table 1 shows the recognition accuracies achieved for PCA only features. For this experimental scenario, we received 85% recognition accuracy for Bayesian-linear classifier, 90% accuracy for Bayesian quadratic, and 95% for 1-NN classifier. Though we expect a 100% accuracy for face-only mode, what we found was that quality of side face images was very poor, resulting in failure to recognize some poor quality faces.

Table 1. PCA with Bayesian Classifiers and 1-Nearest Neighbour Classifier

Classifier Type	Face-Only PCA	Gait-Only PCA	Face-Gait (Feature Fusion)	Face-Gait Score Fusion
Bayesian-linear	85 %	45 %	75 %	65%
Bayesian-quadratic	90 %	50 %	65 %	60%
1-NN classify	95 %	50 %	70 %	55%

Next, we performed experiments for gait only mode, and we achieved a poor recognition accuracy of 45% recognition for Bayesian linear classifier, 50% for Bayesian-quadratic classifier and 50% of 1-NN classifier. Once again, PCA features for gait only mode failed badly because of the inability of PCA technique to capture the gait dynamics of each person. However, when we integrated the face-only information with gait information, the performance improved significantly, resulting in an accuracy of 75%, 65% and 70% for Bayesian-linear, Bayesian-quadratic and 1-NN classifiers respectively.

Further, as can be seen in Table 1, feature level fusion performs better than score level fusion for all three classifiers ascertaining the inherent multimodality in face and gait, which is modeled better with feature-level fusion mode as compared to score-level fusion. For all the experiments in this set we used 40 PCA feature dimensions.

4.2   Recognition Accuracies with PCA-LDA Features

For this set of experiments, we obtained the PCA transformation first and then PCA features were transformed in the LDA space again, training and testing was performed on PCA-LDA vectors, with this, we achieved 100% accuracy for face-only data set. For gait only data set, we achieved a recognition accuracy of 90% for Bayesian-linear, 90% for Bayesian-quadratic, and 80% for 1-NN classifier. Combining the face-gait features in PCA+LDA subspace it was possible to achieve a recognition accuracy of 100% for all three types of classifiers.

Since the face only classifier in PCA-LDA subspace results in 100% accuracy, it would appear that there is no need for fusion with gait features. However, the dimensionality of face only PCA-LDA features was 40 for achieving 100% accuracy, whereas, the dimensionality of features needed to achieve 100^ accuracy was much lesser when face and gait features were fused. We needed 20 features for feature-level fusion and 30 features with score-level fusion to achieve 100% accuracy. As can be seen in Table 2, PCA features in LDA subspace were capable in capturing the person-specific gait variations accurately for all three classifiers. So it was a synergistic fusion, with PCA helpful in reducing the dimensionality and LDA capturing inter-person and intra-person gait associated variations accurately. Another interesting observation was though it is well known in literature, that the score-level fusion results in better performance than feature level fusion, we found that the number of features needed for score fusion is higher (30 as compared to 20 features for feature-level fusion before concatenation). Thus could be because score level fusion does not preserve the inherent multimodality present in face and gait as well as feature-level fusion can do.

Table 2. PCA - LDA with Bayesian Classifiers and 1-Nearest Neighbour Classifier

Name	Face-Only PCA-LDA (40)	Gait-Only PCA-LDA (40)	Face-Gait Feature Fusion (20)	Face-Gait Score Fusion (30)
Bayesian-linear	100%	90%	100%	100%
Bayesian-quadratic	100%	90%	100%	100%
1-NN classify	100%	80%	100%	100%

5 Conclusions and Further Scope

In this paper we propose a novel multimodal Bayesian approach based on PCA-LDA processing for person identification from low resolution surveillance video with cues extracted from gait, face and body biometrics. The experimental evaluation of the proposed scheme on a publicly available database [2] showed that the combined PCA-LDA approach turns out to be a powerful method for capturing the inherent multimodality in walking gait patterns and at the same time discriminating the identity from low resolution video with noisy backgrounds. Further work involves exploring novel methods for identity verification for unconstrained operating environments.

References

1. Murray, M.P., Drought, A.B., Kory, R.C.: Walking pattern of movement. American Journal of Medicine 46(1), 290–332 (1967)
2. Schuldt, C., Laptev, I., Caputo, B.: Recognizing Human Actions: A Local SVM Approach. In: Proceedings ICPR (2004)
3. Bringer, J., Chabanne, H.: Biometric Identification Paradigm Towards Privacy and Confidentiality Protection. In: Nichols, E.R. (ed.) Biometric: Theory, Application and Issues, pp. 123–141 (2011)
4. Zhao, W.-Y., Chellappa, R., Phillips, P.J., Rosenfeld, A.: Face Recognition: A Literature Survey. ACM Computing Surveys 35, 399–458 (2003)
5. Cox, I.J., Ghosn, J., Yianilos, P.N.: Feature-based face recognition using mixture-distance. In: Proc. IEEE Conf. CVPR, pp. 209–216 (1996)
6. Wiskott, L., Fellous, J.M., Kruger, N., Malsburg, C.V.D.: Face Recognition by Elastic Bunch Graph Matching. IEEE Trans. on Pattern Analysis and Machine Intelligence 19, 775–779 (1997)
7. Turk, M., Pentland, A.: Eigenfaces For Recognition. J. Cognitive Neuroscience 3, 71–86 (1991)
8. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. IEEE Trans. Pattern Analysis and Machine Intelligence 19, 711–720 (1997)

9. Zhou, S., Chellappa, R.: Beyond a single still image: Face recognition from multiple still images and videos. In: Face Processing: Advanced Modeling and Methods. Academic Press (2005)
10. Bowyer, K.W., Chang, K., Flynn, P.J.: A survey of approaches and challenges in 3D and multi-modal 3D+2D face recognition. Computer Vision and Image Understanding 101, 1–15 (2006)
11. Kong, S.G., et al.: Recent advances in visual and infrared face recognition - a review. Computer Vision and Image Understanding 97, 103–135 (2005)
12. Ross, A., Jain, A.K.: Information fusion in biometrics. Pattern Recognition Letters 24, 2115–2125 (2003)
13. Jain, A.K., Hong, L., Kulkarni, Y.: A multimodal biometric system using fingerprints, face and speech. In: 2nd Int'l Conf. AVBPA, pp. 182–187 (1999)
14. Wang, Y., Tan, T., Jain, A.K.: Combining face and iris biometrics for identity verification. In: Kittler, J., Nixon, M.S. (eds.) AVBPA 2003. LNCS, vol. 2688, pp. 805–813. Springer, Heidelberg (2003)
15. Chang, K., Bowyer, K.W., Sarkar, S.: Comparison and Combination of Ear and Face Images in Appearance-Based Biometrics. IEEE Trans. PAMI 25, 1160–1165 (2003)
16. Kittler, J., Hatef, M., Duin, R.P.W., Matas, J.: On combining classifiers. IEEE Trans. Pattern Anal. Mach. Intell. 20, 226–239 (1998)
17. Chetty, G., White, M.: Multimedia Sensor Fusion for Retrieving Identity in Biometric Access Control Systems. ACM Transaction on Multimedia Computing, Communications and Applications (Special Issue on Sensor Fusion) (2010)
18. Chetty, G., Wagner, M.: Robust face-voice based speaker identity verification using multilevel fusion. Image and Vision Computing 26, 1249–1260 (2008)
19. Zhou, X., Bhanu, B.: Feature fusion of side face and gait for video-based human identification. In: 2007 Pattern Recognition Society. Elsevier (2007)
20. Hossain, S.M.E., Chetty, G.: Next Generation Identity Verification Based on Face-Gait Biometrics. In: International Conference on Biomedical Engineering and Technology, IPCBEE, vol. 11 (2011)
21. Chakraborthy, R., Rengamani, H., Kumaraguru, P., Rao, R.: The UID Project: Lessons Learned from the Waste and Challenges Identified for India. Cyber Security, Cyber Crime and Cyber Forensics: Application and Perspective (2011)
22. Bringer, J., Chabanne, H.: Biometric Identification Paradigm Towards Privacy and Confidentiality Protection. In: Nichols, E.R. (ed.) Biometric: Theory, Application and Issues, pp. 123–141 (2011)
23. Schuckers, M.E.: Bayesian Method Computational Methods in Biometric Authentication: Statistical Method for performance evolution. Springer, London Limited (2010)
24. Cunningham, P., Delany, S.J.: k-Nearest Neighbor Classifiers, University College Dublin, Padraig.<Literal>Cunningham@ucd.ie</Literal> Dublin Institute of Technology Sarahjane.<Literal>Delany@comp.dit.ie</Literal> Technical Report UCD-CSI-2007-4 (2007)
25. Smith, L.I.: A tutorial on Principal Components Analysis (2002)
26. Sergios, T., Konstantinos, K.: Pattern Recognition. Academic Press (1999)

Robust Control of Nonlinear System Using Difference Signals and Multiple Competitive Associative Nets

Shuichi Kurogi, Hiroshi Yuno, Takeshi Nishida, and Weicheng Huang

Kyusyu Institute of technology, Tobata, Kitakyushu, Fukuoka 804-8550, Japan
{kuro,nishida}@cntl.kyutech.ac.jp
http://kurolab.cntl.kyutech.ac.jp/

Abstract. This paper describes a robust control method using difference signals and multiple competitive associative nets (CAN2s). Using difference signals of a plant to be controlled, the CAN2 is capable of leaning piecewise Jacobian matrices of nonlinear dynamics of the plant. By means of employing the GPC (generalized predictive controller), a robust control method to switch multiple CAN2s to cope with plant parameter change is introduced. We show the effectiveness of the present method via numerical experiments of a crane system.

Keywords: Robust control, Switching of multiple CAN2, Difference signals, Jacobian matrix, Control of nonlinear plant.

1 Introduction

This paper describes a robust control method using difference signals and multiple competitive associative nets (CAN2s). Here, the CAN2 is an artificial neural net for learning an efficient piecewise linear approximation of nonlinear functions by means of competitive and associative schemes[1,2]. The effectiveness is shown in several areas such as control problems, function approximation, rainfall estimation, time series prediction, and so on [3]-[6]. Here, note that the local linear models [7,8] also utilize the piecewise linear approximation, but they use linear models in piecewise regions obtained via the K-nearest neighbors or the SOM (Self-Organizing Map), while the CAN2 utilizes linear models (associative matrices) in the piecewise regions obtained via the competitive learning designed for minimizing the mean square prediction error. Since the K-nearest neighbors and the SOM are for minimizing the distance measures between input vectors and the centers of piecewise regions without any relationship to the prediction error, the CAN2 is expected to show better performance in prediction problems.

As an application of the CAN2, we are developing control methods using CAN2 for learning and utilizing piecewise linear models of nonlinear and time varying plant dynamics [5,6]. Recently in [6], we have presented a method to cope with parameter change of the plant, where we use multiple CAN2s to control the plant with different parameter values. Although the method is shown robust to the parameter change through numerical experiments of the temperature control of RCA cleaning solutions (see [6] for details), we could not have clarified the method analytically.

In the next section, we formulate and analyse the control method using difference signals and multiple CAN2s to show a robust control of nonlinear plant whose parameter may change. And then, in Sect. 3, we examine the effectiveness of the method

through numerical experiments applied to a nonlinear crane system involving several parameter values.

2   Control Method Using Difference Signals and Multiple CAN2s

We formulate and analyse the control method using difference signals and multiple CAN2s to cope with the parameter change.

2.1   Plant Model Using Difference Signals

Suppose a plant to be controlled at a discrete time $j = 1, 2, \cdots$ has the input $u_j^{[p]}$ and the output $y_j^{[p]}$. Here, the superscript "[p]" indicates the variable related to the plant for distinguishing the position of the load, (x, y), shown below. Furthermore, we suppose the dynamics of the plant is given by

$$y_j^{[p]} = f(\boldsymbol{x}_j^{[p]}) + d_j^{[p]}, \tag{1}$$

where $f(\cdot)$ is a nonlinear function which may change slowly in time and $d_j^{[p]}$ represents zero-mean noise with the variance σ_d^2. The input vector $\boldsymbol{x}_j^{[p]}$ of the function consists of the recent input and output of the plant as $\boldsymbol{x}_j^{[p]} \triangleq \left(y_{j-1}^{[p]}, \cdots, y_{j-k_y}^{[p]}, u_{j-1}^{[p]}, \cdots, u_{j-k_u}^{[p]} \right)^T$, where k_y and k_u are positive integers, and the dimension of $\boldsymbol{x}_j^{[p]}$ is $k = k_y + k_u$. Then, for the difference signals $\Delta y_j^{[p]} \triangleq y_j^{[p]} - y_{j-1}^{[p]}$, $\Delta u_j^{[p]} \triangleq u_j^{[p]} - u_{j-1}^{[p]}$, and $\Delta \boldsymbol{x}_j^{[p]} \triangleq \boldsymbol{x}_j^{[p]} - \boldsymbol{x}_{j-1}^{[p]}$, we have the relationship $\Delta y_j^{[p]} \simeq \boldsymbol{J}_f \Delta \boldsymbol{x}_j^{[p]}$ for small $\|\Delta \boldsymbol{x}_j^{[p]}\|$, where $\boldsymbol{J}_f = \partial f(\boldsymbol{x})/\partial \boldsymbol{x} \big|_{\boldsymbol{x}=\boldsymbol{x}_{j-1}^{[p]}}$ indicates the Jacobian matrix (row vector). If $\boldsymbol{J}_f$ does not change for a while around the time j, then we can predict $\Delta y_{j+l}^{[p]}$ by

$$\widehat{\Delta y}_{j+l}^{[p]} = \boldsymbol{J}_f \widetilde{\Delta \boldsymbol{x}}_{j+l}^{[p]} \tag{2}$$

for $l = 1, 2, \cdots$, recursively. Here, the elements of $\widetilde{\Delta \boldsymbol{x}}_{j+l}^{[p]} = (\widetilde{\Delta y}_{j+l-1}^{[p]}, \cdots, \widetilde{\Delta y}_{j+l-k_y}^{[p]}, \widetilde{\Delta u}_{j+l-1}^{[p]}, \cdots, \widetilde{\Delta y}_{j+l-k_u}^{[p]})^T$ are the past and the predictive input and output given by

$$\widetilde{\Delta y}_{j+m}^{[p]} = \begin{cases} \Delta y_{j+m}^{[p]} & \text{for } m < 1 \\ \widehat{\Delta y}_{j+m}^{[p]} & \text{for } m \geq 1 \end{cases} \quad \text{and} \quad \widetilde{\Delta u}_{j+m}^{[p]} = \begin{cases} \Delta u_{j+m}^{[p]} & \text{for } m < 0 \\ \widehat{\Delta u}_{j+m}^{[p]} & \text{for } m \geq 0. \end{cases} \tag{3}$$

Here, see Sect. 2.3 for the predictive input $\widehat{\Delta u}_{j+m}^{[p]}$ $(m \geq 0)$. Then, we have the prediction of the plant output from the predictive difference signals as

$$\widehat{y}_{j+l}^{[p]} = y_j^{[p]} + \sum_{m=1}^{l} \widehat{\Delta y}_{j+m}^{[p]}. \tag{4}$$
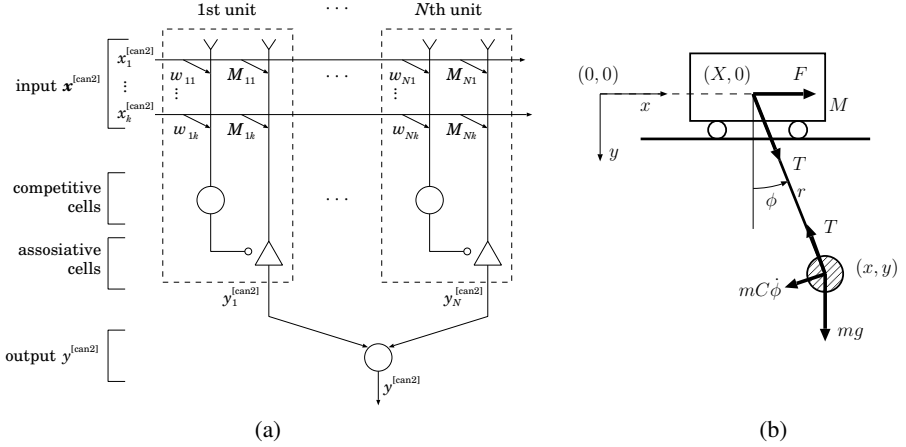
Fig. 1. Schematic diagram of (a) the CAN2 and (b) the overhead crane system

2.2 CAN2 Using Difference Signals and Relation to Parameter Change

A CAN2 has N units. The ith unit has a weight vector $\boldsymbol{w}_i \triangleq (w_{i1}, \cdots, w_{ik})^T \in \mathbb{R}^{k \times 1}$ and an associative matrix (row vector) $\boldsymbol{M}_i \triangleq (M_{i1}, \cdots, M_{ik}) \in \mathbb{R}^{1 \times k}$ for $i \in I = \{1, 2, \cdots, N\}$ (see Fig.1(a)). In order for the CAN2 to learn a given dataset $D^n = \{(\boldsymbol{x}_j^{[\mathrm{p}]}, y_j^{[\mathrm{p}]}) \mid j = 1, 2, \cdots, n\}$ obtained from the plant, we feed the input and output of the CAN2 as $(\boldsymbol{x}^{[\mathrm{can2}]}, y^{[\mathrm{can2}]}) = (\Delta \boldsymbol{x}_j^{[\mathrm{p}]}, \Delta y_j^{[\mathrm{p}]})$. Then, the CAN2 after learning (see [4] for the learning method) approximates the output $\Delta y_j^{[\mathrm{p}]} = f(\Delta \boldsymbol{x}_j^{[\mathrm{p}]})$ by

$$\widehat{\Delta y}_j^{[\mathrm{p}]} = \boldsymbol{M}_c \Delta \boldsymbol{x}_j^{[\mathrm{p}]}, \tag{5}$$

where the index of the unit, c, is selected by

$$c = \operatorname*{argmin}_{i \in I} \|\Delta \boldsymbol{x}_j^{[\mathrm{p}]} - \boldsymbol{w}_i\|^2. \tag{6}$$

Now, let us examine Eqs.(5) and (6). From Eq.(2) and Eq.(5), we can see that the associative matrix approximates the Jacobian matrix, or $\boldsymbol{M}_c \simeq \boldsymbol{J}_f$. Furthermore, Eqs.(5) and (6) indicate an assumption that the associative or the Jacobian matrix can be identified by $\Delta \boldsymbol{x}_j^{[\mathrm{p}]}$, although it depends on the original signal $\boldsymbol{x}_{j-1}^{[\mathrm{p}]}$ from the definition $\boldsymbol{J}_f = \partial f(\boldsymbol{x})/\partial \boldsymbol{x} \mid_{\boldsymbol{x} = \boldsymbol{x}_{j-1}^{[\mathrm{p}]}}$. However, when $\boldsymbol{J}_f$ does not change for a while and $\Delta \boldsymbol{x}_{j-m}^{[\mathrm{p}]}$ $(m = 1, 2, \cdots)$ involves k $(= \dim \Delta \boldsymbol{x}_j^{[\mathrm{p}]})$ linearly independent vectors, we can derive $\boldsymbol{J}_f = \left[\Delta \boldsymbol{x}_{j-1}^{[\mathrm{p}]}, \cdots, \Delta \boldsymbol{x}_{j-k}^{[\mathrm{p}]}\right]^{-1} \left[\Delta y_{j-1}^{[\mathrm{p}]}, \cdots, \Delta y_{j-k}^{[\mathrm{p}]}\right]$ from $\Delta y_{j-m}^{[\mathrm{p}]} = \boldsymbol{J}_f \Delta \boldsymbol{x}_{j-m}^{[\mathrm{p}]}$ for $m = 1, 2, \cdots, k$. This indicates that $\boldsymbol{J}_f$ can be identified by $2k$ - dimensional vector, $\Delta \boldsymbol{x}_j'^{[\mathrm{p}]} = (\Delta y_{j-1}^{[\mathrm{p}]}, \cdots, \Delta y_{j-k-k_y}^{[\mathrm{p}]}, \Delta u_{j-1}^{[\mathrm{p}]}, \cdots, \Delta u_{j-k-k_u}^{[\mathrm{p}]})$ which involves all elements in $\left[\Delta \boldsymbol{x}_{j-1}^{[\mathrm{p}]}, \cdots, \Delta \boldsymbol{x}_{j-k}^{[\mathrm{p}]}\right]$. Thus, the relation from $\Delta \boldsymbol{x}_j'^{[\mathrm{p}]}$ to $\Delta y_j^{[\mathrm{p}]}$

becomes a single valued function $\Delta y_j^{[\mathrm{p}]} = \boldsymbol{J}_f \Delta \boldsymbol{x}_j^{[\mathrm{p}]} = g(\Delta \boldsymbol{x}_j'^{[\mathrm{p}]})$. Then, considering the linear approximation $\Delta y_j^{[\mathrm{p}]} = g(\Delta \boldsymbol{x}_j'^{[\mathrm{p}]}) \simeq \boldsymbol{J}_g \Delta \boldsymbol{x}_j'^{[\mathrm{p}]}$, the relation from $\Delta \boldsymbol{x}_j'^{[\mathrm{p}]}$ to $\boldsymbol{J}_g$ is a vector-valued function. Therefore, the Jacobian $\boldsymbol{J}_f$ for the enlarged vector $\Delta \boldsymbol{x}_j^{[\mathrm{p}]} = (\Delta y_{j-1}^{[\mathrm{p}]}, \cdots, \Delta y_{j-k_y'}^{[\mathrm{p}]}, \Delta u_{j-1}^{[\mathrm{p}]}, \cdots, \Delta u_{j-k_u'}^{[\mathrm{p}]})$ with $k_y' = k + k_y$ and $k_u' = k + k_u$ is a vector-valued function of $\Delta \boldsymbol{x}_j^{[\mathrm{p}]}$, or $\boldsymbol{J}_f = \boldsymbol{J}_f(\Delta \boldsymbol{x}_j^{[\mathrm{p}]})$. Thus, let's use this enlarged vector $\Delta \boldsymbol{x}_j^{[\mathrm{p}]}$ below from here, so that the control using $\Delta y_{j+l}^{[\mathrm{p}]}$ predicted from $\Delta \boldsymbol{x}_j^{[\mathrm{p}]}$ is expected to be robust to the parameter change of the plant because parameter values are reflected by $\boldsymbol{J}_f$ which can be identified by $\Delta \boldsymbol{x}_j^{[\mathrm{p}]}$.

Moreover, note that the learning and the prediction using $(\Delta \boldsymbol{x}_j^{[\mathrm{p}]}, \Delta y_j^{[\mathrm{p}]})$ is more efficient than those using $(\boldsymbol{x}_j^{[\mathrm{p}]}, y_j^{[\mathrm{p}]})$ as shown in [5]. Namely, for learning and predicting the relationship $y_j^{[\mathrm{p}]} = f(\boldsymbol{x}_j^{[\mathrm{p}]}) \simeq b_j + \boldsymbol{J}_f \boldsymbol{x}_j^{[\mathrm{p}]}$, the CAN2 using the original signals $(\boldsymbol{x}^{[\mathrm{can2}]}, y^{[\mathrm{can2}]}) = (\boldsymbol{x}_j^{[\mathrm{p}]}, y_j^{[\mathrm{p}]}) \simeq (\boldsymbol{x}_j^{[\mathrm{p}]}, b_j + \boldsymbol{J}_f \boldsymbol{x}_j^{[\mathrm{p}]})$ has to learn the data for all bias b_j, while the CAN2 using the difference signals $(\boldsymbol{x}^{[\mathrm{can2}]}, y^{[\mathrm{can2}]}) = (\Delta \boldsymbol{x}_j^{[\mathrm{p}]}, \Delta y_j^{[\mathrm{p}]}) = (\Delta \boldsymbol{x}_j^{[\mathrm{p}]}, \boldsymbol{J}_f \Delta \boldsymbol{x}_j^{[\mathrm{p}]})$ does not have to learn b_j and the prediction for any bias b_j can be done by Eq.(4).

2.3   GPC for Difference Signals

The GPC (Generalized Predictive Control) is an efficient method for obtaining the predictive input $\widehat{u}_j^{[\mathrm{p]}}$ which minimizes the control performance index:

$$J = \sum_{l=1}^{N_y} (r_{j+l}^{[\mathrm{p}]} - \widehat{y}_{j+l}^{[\mathrm{p}]})^2 + \lambda_u \sum_{l=1}^{N_u} (\widehat{\Delta u}_{j+l-1}^{[\mathrm{p}]})^2, \tag{7}$$

where $r_{j+l}^{[\mathrm{p}]}$ and $\widehat{y}_{j+l}^{[\mathrm{p}]}$ are desired and predictive output, respectively. The parameters N_y, N_u and λ_u are constants to be designed for the control performance. We obtain $\widehat{u}_j^{[\mathrm{p}]}$ by means of the GPC method as follows; the CAN2 at a discrete time j can predict $\Delta y_{j+l}^{[\mathrm{p}]}$ by Eq.(2) and then $\widehat{y}_{j+l}^{[\mathrm{p}]}$ by Eq.(4). Then, owing to the linearity of these equations, the above performance index is written as

$$J = \|\boldsymbol{r}^{[\mathrm{p}]} - \boldsymbol{G} \Delta \boldsymbol{u}^{[\mathrm{p}]} - \overline{\boldsymbol{y}}^{[\mathrm{p}]}\|^2 + \lambda_u \|\widehat{\Delta \boldsymbol{u}}\|^2 \tag{8}$$

where $\boldsymbol{r}^{[\mathrm{p}]} = \left(r_{j+1}^{[\mathrm{p}]}, \cdots, r_{j+N_y}^{[\mathrm{p}]} \right)^T$ and $\widehat{\Delta \boldsymbol{u}}^{[\mathrm{p}]} = \left(\widehat{\Delta u}_j^{[\mathrm{p}]}, \cdots, \widehat{\Delta u}_{j+N_u-1}^{[\mathrm{p}]} \right)^T$. Furthermore, $\overline{\boldsymbol{y}}^{[\mathrm{p}]} = \left(\overline{y}_{j+1}^{[\mathrm{p}]}, \cdots, \overline{y}_{j+N_y}^{[\mathrm{p}]} \right)^T$ and $\overline{y}_{j+l}^{[\mathrm{p}]}$ is the natural response $\widehat{y}_{j+l}^{[\mathrm{p}]}$ of the system Eq.(1) for the null incremental input $\widehat{\Delta u}_{j+l}^{[\mathrm{p}]} = 0$ for $l \geq 0$. Here, we actually have $\overline{y}_{j+l}^{[\mathrm{p}]} = y_j^{[\mathrm{p}]} + \sum_{m=1}^{l} \overline{\Delta y}_{j+m}^{[\mathrm{p}]}$ from Eq.(4), where $\overline{\Delta y}_{j+l}^{[\mathrm{p}]}$ denotes the natural response of the difference system of Eq.(2) with $\boldsymbol{J}_f$ replaced by $\boldsymbol{M}_c$. The ith column and the jth row of the matrix $\boldsymbol{G}$ is given by $G_{ij} = g_{i-j+N_1}$, where g_l for $l = \cdots, -2, -1, 0, 1, 2, \cdots$ is the unit step response $y_{j+l}^{[\mathrm{p}]}$ of Eq.(4) for $\widehat{y}_{j+l}^{[\mathrm{p}]} = \widehat{u}_{j+l}^{[\mathrm{p}]} =$

0 $(l < 0)$ and $\widehat{u}^{[\mathrm{p}]}_{j+l} = 1 (l \geq 0)$. It is easy to derive that the unit response g_l of Eq.(4) is obtained as the impulse response of Eq.(2). Then, we have $\widehat{\Delta u}^{[\mathrm{p}]}$ which minimizes J by $\widehat{\Delta u}^{[\mathrm{p}]} = (\boldsymbol{G}^T \boldsymbol{G} + \lambda_u \boldsymbol{I})^{-1} \boldsymbol{G}^T (\boldsymbol{r}^{[\mathrm{p}]} - \overline{\boldsymbol{y}}^{[\mathrm{p}]})$, and then we have $\widehat{u}^{[\mathrm{p}]}_j = u^{[\mathrm{p}]}_{j-1} + \widehat{\Delta u}^{[\mathrm{p}]}_j$.

2.4 Iterations of Control and Learning

In order to improve the control performance, we execute iterations of the following phases.

(i) **control phase:** Control the plant by some default control schedule at the first iteration, and by the above GPC using the CAN2 after the phase (ii) otherwise.
(ii) **learning phase:** Using the dataset $D^n = \{(\boldsymbol{x}^{[\mathrm{p}]}_j, y^{[\mathrm{p}]}_j | j = 1, 2, \cdots, n)\}$ obtained from the control phase, apply the batch learning method to the CAN2 [4].

2.5 Switching Multiple CAN2s to Cope with Parameter Change

To cope with parameter change of the plant, we may use a CAN2 for learning a huge dataset of $(\Delta \boldsymbol{x}^{[\mathrm{p}]}_j, y^{[\mathrm{p}]}_j)$ produced from the plants for different parameter values. However, to avoid time-consuming learning of the huge dataset, we employ the following steps using multiple CAN2s, each of which, we denote $\mathrm{CAN2}^{[\theta_s]}$, is the CAN2 after learning the training dataset for parameter θ_s and $s \in S = \{1, 2, \cdots\}$.

step 1: At the time j in the control phase, select the unit for each $\mathrm{CAN2}^{[\theta_s]}$ by Eq.(6), or $c^{[s]} = \underset{i \in I}{\mathrm{argmin}} \| \Delta \boldsymbol{x}^{[\mathrm{p}]}_j - \boldsymbol{w}^{[s]}_i \|^2$, where $\boldsymbol{w}^{[s]}_i$ $(i \in I)$ are the weight vectors of $\mathrm{CAN2}^{[\theta_s]}$.

step 2: Select the s^*th CAN2 which has the minimum MSE (mean square prediction error) for the recent N_e outputs,

$$s^* = \underset{s \in S}{\mathrm{argmin}} \frac{1}{N_e} \sum_{l=0}^{N_e-1} \| \Delta y^{[\mathrm{p}]}_{j-l} - \boldsymbol{M}_{c^{[s]}} \Delta \boldsymbol{x}^{[\mathrm{p}]}_{j-l}) \|^2, \qquad (9)$$

where $\boldsymbol{M}_{c^{[s]}}$ is the $c^{[s]}$th associative matrix of $\mathrm{CAN2}^{[\theta_s]}$.

Note that **step 2** is necessary because $c^{[s]}$ obtained by **step 1** indicates the optimal unit only for the Voronoi partition of the sth CAN2 and different CAN2 has different Voronoi partition. For evaluate the fitness to the recent data, the above MSE seems to be a reasonable criterion.

3 Numerical Experiments of Crane System

In order to examine the effectiveness of the present method, we execute numerical experiments of the following crane system.

3.1   Overhead Traveling Crane System

We consider the overhead traveling crane system shown in Fig.1(b). From the figure, we have the position and motion equations given by

$$(x, y) = (X + r \sin \phi, r \cos \phi) \tag{10}$$

$$m(\ddot{x}, \ddot{y}) = (-T \sin \phi - mC\dot{\phi} \cos \phi, mg - T \cos \phi - mC\dot{\phi} \sin \phi) \tag{11}$$

$$M\ddot{X} = F + T \sin \phi \tag{12}$$

where (x, y) and m are the position and the weight of the suspended load, $(X, 0)$, M and F are the position, weight and driving force of the trolley, r and ϕ are the length and the angle of the rope, T is the tension of the rope, and C is the viscous damping coefficient. From Eqs.(10) and (11), we have the nonlinear second order differential equation of ϕ given by $r\ddot{\phi} + (C + 2\dot{r})\dot{\phi} + g \sin \phi + \ddot{X} \cos \phi = 0$. Thus, with Eq.(12), the transition of the state $\boldsymbol{x} = \left(\phi, \dot{\phi}, X, \dot{X} \right)^T$ is given by

$$\dot{\boldsymbol{x}} = h(\boldsymbol{x}) = \begin{bmatrix} \dot{\phi} \\ -\dfrac{C + 2\dot{r}}{r}\dot{\phi} - \dfrac{g}{r}\sin \phi - \dfrac{F + T\sin \phi}{rM}\cos \phi \\ \dot{X} \\ \dfrac{F + T\sin x_1}{M} \end{bmatrix}, \tag{13}$$

where $T = m\sqrt{(\ddot{x} + C\dot{\phi} \cos \phi)^2 + (\ddot{y} - g + C\dot{\phi} \sin \phi)^2}$ is also a function of $\boldsymbol{x}$. The control objective is to move the horizontal position of the load, $x = X + r \sin \phi$, to a destination position x_d by means of operating F.

3.2   Parameter Settings

The parameter values of the crane system are set as follows: the trolley weight $M = 100$kg, the damping coefficient $C = 0.5$m/s, the maximum driving force $F_{\max} = 20$N, and we have examined the robustness to the rope length r from 2 to 10m, and the load weight from 10 to 100kg. We obtain the discrete signals by $u_j^{[\text{p}]} = F(jT_v)$ and $y_j^{[\text{p}]} = x(jT_v)$ with $T_v = 0.5$s. Here, we use the virtual sampling method shown in [3], where the discrete model is obtained with the virtual sampling period T_v while the observation and operation are executed with shorter actual sampling period $T_a = 0.01$s. We use $k_y' = k_u' = 4$ for the enlarged input vector $\Delta \boldsymbol{x}_j^{[\text{p}]}$, and $N_y = 20$, $N_u = 1$ and $\lambda_u = 0.01$ for the GPC. We used $N_e = 50$ samples for switching multiple CAN2s, where we used the actual sampling period T_a for the discrete time $j-l$ in Eq.(9), thus the period of time for N_e samples indicates $N_e T_a = 0.5$s. Let crane$^{[\theta]}$ denote the crane with the parameter θ. Especially, let $\theta = \theta_s$ for $s = 1, 2, 3, 4$ denote $\theta = (r, m) = (2, 10)$, $(2, 100)$, $(10, 10)$ and $(10, 100)$, respectively, which are used for training CAN2s. Let CAN2$^{[\theta_s]}$ denote the CAN2 made through 20 iterations of control and learning phases for crane$^{[\theta_s]}$. We use $N = 5$ units for each CAN2. Let CAN2$^{[\theta_S]}$ denote the set of all CAN2$^{[\theta_s]}$ for $s \in S = \{1, 2, 3, 4\}$ used for the switching controller.
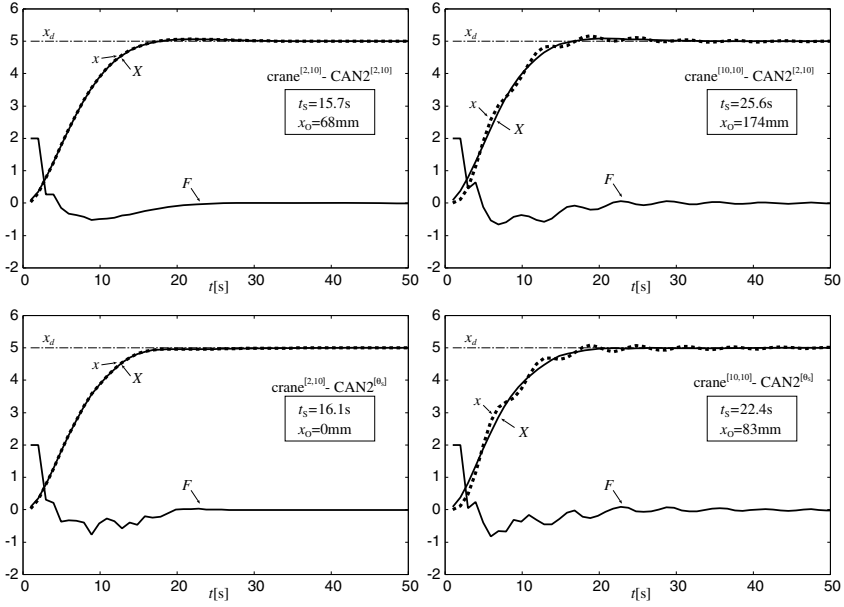
Fig. 2. Resultant time courses of x, X and F for the controller using single CAN2[2,10] (top) and multiple CAN2[θ_S] (bottom), and the crane with the parameter $\theta = (r, m) = (2, 10)$ (left side) and $\theta = (10, 10)$ (right side). The unit of x, X and x_d is [m], and that of F is [10N].

3.3 Results and Remarks

Examples of resultant time courses of x, X and F are shown in Fig. 2, where we can see that the controller using single CAN2[2,10] has controlled not crane[10,10] but crane[2,10] with allowable overshoot less than 100mm. On the other hand, the switching controller using multiple CAN2[θ_S] has controlled both cranes with allowable overshoot.

A summary of the resultant settling time and overshoot by the controllers applied to the crane for several parameter values is shown in Table 1. We can see that the controller using a single CAN2[θ_s] has controlled crane[θ_s] with allowable overshoot, but has not always controlled other crane[θ] for $\theta \neq \theta_s$. However, the switching controller using multiple CAN2[θ_S] has controlled every crane[θ] for $\theta = \theta_s$ ($s = 1, 2, 3, 4$) as well as $\theta = (r, m)$ not involved in θ_S with allowable overshoot.

The above robust control to the parameter change is owing to the switching method shown in Sect. 2.5 which selects the best $M_{c[s*]} \simeq J_f$ for predicting $\Delta y_{j+l}^{[p]}$ at each time j, which is capable because J_f is a function of $\Delta x_j^{[p]}$ regardless of θ as shown in Sect. 2.2. Furthermore, one of the advantages of the present method is that the robustness to additional parameter value can be incrementally added by means of employing additional CAN2.

Table 1. Settling time and overshoot for parameter change. The parameter θ_s $(s = 1, 2, 3, 4, S)$ in the second row from the top indicates CAN2$^{[\theta_s]}$, and θ_s $(s = 1, 2, 3, 4)$ in the leftmost column indicates the controlled crane$^{[\theta_s]}$. The underlined figure shows the result by the controller using CAN2$^{[\theta_s]}$ applied to crane$^{[\theta_s]}$. The thick figure indicates the best (smallest) settling time and overshoot in each row.

crane	Settling time t_S(s)					Overshoot x_O(mm)				
$\theta = (r, m)$	θ_1	θ_2	θ_3	θ_4	θ_S	θ_1	θ_2	θ_3	θ_4	θ_S
$\theta_1 = (2, 10)$	<u>15.7</u>	21.1	**15.2**	21.3	16.1	<u>68</u>	105	26.8	397	**0**
$(2, 55)$	**14.8**	20.2	26.8	21.5	16.9	77	103	6	288	**0**
$\theta_2 = (2, 100)$	21.6	<u>14.9</u>	21.2	**13.5**	14.0	185	<u>63</u>	**0**	100	73
$(6, 10)$	**14.7**	20.0	26.3	16.4	17.9	91	109	358	69	**6**
$(6, 55)$	**14.2**	22.7	27.0	15.4	15.3	94	158	**0**	17	48
$(6, 100)$	**20.6**	27.5	34.5	24.7	21.1	126	220	462	12	**0**
$\theta_3 = (10, 10)$	25.6	25.8	<u>23.6</u>	24.2	**22.4**	174	204	<u>**36**</u>	251	83
$(10, 55)$	14.1	15.0	28.9	15.5	**13.8**	81	21	384	**7**	50
$\theta_4 = (10, 100)$	20.4	22.0	20.6	**14.2**	14.2	141	162	9	**5**	43
average	18.0	21.0	24.9	18.5	**16.9**	115	127	139	127	**34**

4   Conclusion

We have presented a robust control method using multiple CAN2s and difference signals. The CAN2 using difference signals of a plant is shown to lean Jacobian matrices of nonlinear dynamics of a plant. We have introduced the switching method of multiple CAN2s to cope with plant parameter change after showing the GPC using difference signals to control the plant. Via numerical experiments of a crane system, we have shown the effectiveness of the present method. We would like to derive necessary and/or sufficient conditions for this method in our future research study.

References

1. Kohonen, T.: Associative Memory. Springer, Heidelberg (1977)
2. Ahalt, A.C., Krishnamurthy, A.K., Chen, P., Melton, D.E.: Competitive learning algorithms for vector quantization. Neural Networks 3, 277–290 (1990)
3. Kurogi, S., Nishida, T., Sakamoto, T., Itoh, K., Mimata, M.: A simplified competitive associative net and a model-switching predictive controller for temperature control of chemical solutions. In: Proc. of ICONIP 2000, pp. 791–796 (2000)
4. Kurogi, S., Sawa, M., Ueno, T., Fuchikawa, Y.: A batch learning method for competitive associative net and its application to function approximation. In: Proc. of SCI 2004, vol. 5, pp. 24–28 (2004)
5. Kurogi, S., Koshiyama, Y., Yuno, Y.: Temperature control of RCA cleaning solutions via model switching predictive control using first-order difference signals and bagging CAN2. Transaction of SICE 45(6), 305–312 (2009)

6. Kurogi, S., Yuno, H., Koshiyama, Y.: A method to switch multiple CAN2s for variable initial temperature in temperature control of RCA cleaning solutions. In: Leung, C.S., Lee, M., Chan, J.H. (eds.) ICONIP 2009. LNCS, vol. 5864, pp. 56–64. Springer, Heidelberg (2009)
7. Farmer, J.D., Sidorowich, J.J.: Predicting chaotic time series. Phys. Rev. Lett. 59, 845–848 (1987)
8. Ritter, H., Martinetz, T., Schulten, K.: Neural computation and self-orgaizing maps. Addison and Wesley (1992)

Selective Track Fusion

Li Xu[1,2], Peijun Ma[1], and Xiaohong Su[1]

[1] School of Computer Science and Technology, Harbin Institute of Technology,
Harbin 150001, China
[2] College of Computer Science and Technology, Harbin Engineering University,
Harbin 150001, China
`xuli_hit@126.com, {ma,sxh}@hit.edu.cn`

Abstract. In this paper, the relationship between the fusion result and the number of sensor tracks taking part in fusion is investigated, which reveals that it may be better to fuse many instead of all of the sensor tracks at hand. This result is interesting because at present, most approaches fuse all the available sensor tracks and treat all sensor data equally without regard of their different quality and different contribution to the system tracks. Then, in order to show that the appropriate sensor tracks for a fusion can be effectively selected from a set of available sensor tracks, an approach named STF is presented. STF is based on a two-stage paradigm of heuristic function construction and track state estimation fusion. The outliers in the tracks are eliminated by the orthogonal polynomial regression method at first. Then heuristic function is constructed by evaluating the quality of each track using grey correlation degree. Last, the track state estimation fusion is guided by the heuristic function, in which an optimal number of tracks are fused. In addition, the paper discusses its implementation in the multi-sensor and multi-target environment. The effectiveness and the superiority of STF are verified in experiment.

Keywords: Track fusion, Heuristics, Multi-sensor, Multi-target, Grey correlation degree.

1 Introduction

Nowadays, the research of track fusion has been a very hot spot. Some theoretical and practical problems are international puzzles, but a lot of famous scholars still devote themselves to the research [1,2]. At present, Measurement Fusion (MF), Simple Fusion (SF) and Weighted Covariance Fusion (WCF) are the most representative track fusion algorithms. MF possesses the advantage of simple idea and small computational cost, but lower accuracy. SF was proposed by Siger and has a notable advantage, high efficiency. However, the accuracy of SF is still not high. WCF was proposed by Bar-Shalom, which is also a classical algorithm. It is characterized by high accuracy, but heavy computational cost. Until now, some performances of them are still the objectives pursued. The accuracy required by the system is variable, so sometimes the simple fusion algorithm can gain the same result as the complex one.

For such considerations, Beugnon, Singh, Llinas and Saha proposed an adaptive track fusion algorithm [3]. In 2006, Li, Cheng, Zhang and Shen added the feedback structure on the basis of the adaptive track fusion algorithm [4]. The fusion center feeds back the state estimation and the error covariance to the local nodes as the initial value of the filter in the next interval to improve the filtering accuracy. In recent years, the scholars put forward many solutions to the track fusion problems under different environment. Chen and Bar-Shalom modify the existing track fusion algorithm accounting for the model mismatch among some of the local tracks [5]. Watson and Rice develop the solution for the fusion of multiple tracks from an arbitrary number of asynchronous measurements with a low complexity [6]. Chang, Chong and Mori focus on scalable fusion algorithms and conduct analytical performance evaluation under different operating conditions [7]. There are also many other approaches for track fusion. Examples are as follows. Yuan, Dong and Wang propose a fusion algorithm based on an adaptive neuro-fuzzy inference system, which combines the merits of fuzzy logic and neural network [8]. Duan and Li propose two optimal distributed fusion algorithms by taking linear transformation of the raw measurements of each sensor [9]. Hu, Duan and Zhou propose a distributed asynchronous fusion algorithm by reconstructing the optimal centralized fusion result [10]. These works promote the research of the track fusion problem.

No matter what kind of fusion technology we adopt, the selection of reliable data is the precondition of a successful fusion. The reliability of data is important to the validity of a fusion system. So before a track fusion, what is a high-quality track and how many tracks should take part in fusion should be answered. In response to the above problems, an approach named Selective Track Fusion (STF) is presented.

2   System Description

Throughout this paper, the research is with the assumption that M sensors observe T targets in clutter. The observation obtained in the discrete time is made up of several measurements, of which some are from the targets and others are from the clutter.

Let $x(k)$ be the state vector at time k. The target model can be expressed as:

$$x(k+1) = F(k)x(k) + G(k)u(k) + v(k) \tag{1}$$

where $F(k)$ is a state transition matrix, $G(k)$ is an input control matrix, $u(k)$ is a mobile acceleration input matrix, $v(k)$ is a discrete-time white noise sequence, and $E[v(k)] = 0$.

The measurement equation of sensor i can be expressed as:

$$Z_i(k) = H(k)x(k) + w_i(k) \tag{2}$$

where $H(k)$ is an observation matrix, and $w_i(k)$ is Gauss observation noise with zero mean and variance $R_i(k)$.

3   Outlier Elimination

In order to improve the quality of the sensor tracks, we eliminate outliers at first.

The analog signal parameters of track possess non-linear characteristics. In a closed interval, it can be described by a p-th polynomial of t:

$$x(k) = \alpha_0 + \alpha_1 t_k + a_2 t_k^2 + \cdots + a_p t_k^p + \varepsilon_k, \quad k = 0,1,2,\cdots,n \qquad (3)$$

Suppose the sampling interval is I and $t_0 = 0$, then the k-th measurement time is $t_k = t_0 + kI = kI$. Substitute t_k into (3), we have:

$$x(k) = \alpha_0 + \alpha_1(kI) + a_2(kI)^2 + \cdots + a_p(kI)^p + \varepsilon_k \qquad (4)$$

Let $\beta_0 = \alpha_0$, $\beta_1 = \alpha_1 I$, $\beta_2 = \alpha_2 I^2$, $\cdots$, $\beta_p = \alpha_p I^p$. Substitute β_i into (4), we have:

$$x(k) = \beta_0 + \beta_1 k + \beta_2 k^2 + \cdots + \beta_p k^p + \varepsilon_k, \quad k = 0,1,2,\cdots,n \qquad (5)$$

Take the following orthogonal polynomials:

$$
\begin{aligned}
&\phi_0(k) = 1 \\
&\phi_1(k) = k - \bar{k} \\
&\phi_2(k) = (k - \bar{k})^2 - \frac{n^2 - 1}{12} \\
&\phi_3(k) = (k - \bar{k})^3 - \frac{3n^2 - 7}{20}(k - \bar{k}) \\
&\phi_4(k) = (k - \bar{k})^4 - \frac{3n^2 - 13}{14}(k - \bar{k})^2 + \frac{3(n^2 - 1)(n^2 - 9)}{560} \\
&\cdots\cdots \\
&\phi_{p+1}(k) = \phi_1(k)\phi_p(k) - \frac{p^2(n^2 - p^2)}{4(4p^2 - 1)}\phi_{p-1}(k)
\end{aligned}
\qquad (6)
$$

where, $\bar{k} = \dfrac{n+1}{2}$.

According to the table of orthogonal polynomial, we can calculate the coefficient of the regression equation:

$$\beta_i' = \frac{\sum\limits_{k=1}^{n} \phi_i(k)x(k)}{\sum\limits_{k=1}^{n} \phi_i^2(k)} \quad i = 0,1,2,\cdots,p \tag{7}$$

And the regression equation is:

$$\hat{x}(k) = \beta_0' + \beta_1'\phi_1(k) + \beta_2'\phi_2(k) + \cdots + \beta_p'\phi_p(k) \tag{8}$$

Based on the theory of statistics, the random error variance is:

$$\hat{\sigma}^2 = \frac{1}{n}\sum\limits_{j=0}^{n}(x(j) - \hat{x}(j))^2 \tag{9}$$

We exploit the orthogonal regression mathematical models to estimate the state value of the sampling point and then compare it with the corresponding measurement value. If the difference is less than $5\hat{\sigma}$, namely: $|x(t) - \hat{x}(t)| < 5\hat{\sigma}$, then the value is reasonable. If not, the point is regarded as an outlier which should be eliminated and be replaced by $\hat{x}(t)$.

4   Heuristics Function Construction

In the multi-sensor environment, there is a large amount of redundant information and the contribution of every sensor track to system track is different. However, most existing multi-sensor systems treat all sensor tracks equally. A track quality evaluation, as done here, is a useful preliminary step to design Heuristics Function.

Suppose M sensors observe the targets at the same time, and $x_i = \{(x_i(1), x_i(2), \cdots, x_i(n)) \mid i = 1, \cdots, M\}$ is the output data from the sensor i. Then the expectation $\overline{x}(j)$ can be expressed as:

$$\overline{x}(j) = \frac{1}{M}\sum\limits_{i=1}^{M} x_i(j), \quad j = 1,2,\cdots,n \tag{10}$$

The grey correlation degree can describe the quality of sensor track effectively. Based on the grey theory, the quality of a track can be defined as follows:

$$h(x_i) = \frac{1}{n}\sum\limits_{j=1}^{n} \frac{\min\limits_{1\le i\le M}\min\limits_{1\le j\le n}|\overline{x}(j) - x_i(j)| + \zeta \max\limits_{1\le i\le M}\max\limits_{1\le j\le n}|\overline{x}(j) - x_i(j)|}{|\overline{x}(j) - x_i(j)| + \zeta \min\limits_{1\le i\le M}\min\limits_{1\le j\le n}|\overline{x}(j) - x_i(j)|} \tag{11}$$

where ζ is the environment correlation factor, and $\zeta = 0.5$ in this paper. For a track, the larger its grey correlation degree is, the better its quality is.

The heuristic function *Select* (*C*) selects a track based on its quality from the set of sensor tracks *C*. *Select* (*C*) is defined as:

$$Select(C) = \arg\max\{h(x_i) \mid x_i \in C\} \tag{12}$$

5   Selective Track Fusion

STF selects the high-quality tracks for the sequential fusion based on the heuristic function.

5.1   The Basic Idea of STF

For presentation convenience, the sensor tracks are numbered by the quality of the tracks. Initially, let $i = 1$, then the algorithm selects two highest-quality sensor tracks S_1 and S_2 for fusion in accordance with the heuristic function, and the fusion track S_1' is acquired. If $h(S_1') < h(S_1)$, the algorithm is over and S_1 is the ultimate system track. Otherwise, * let $i = i + 1$, and choose S_{i+1} based on the heuristic function, then fuse S_{i+1} and S_{i-1}', and the fusion track S_i' is acquired. Repeat * until $h(S_i') < h(S_{i-1}')$ or all the tracks have taken part in the fusion. The algorithm takes S_{i-1}' as the ultimate system track.

5.2   Track State Estimation Fusion

The fusion center fuses the tracks from the local sensors to acquire the system tracks. Here, we exploit SF strategy.

The global state estimation at k is:

$$\hat{x}_{\text{STF}}(k) = P_j(k)[P_i(k) + P_j(k)]^{-1} x_i(k) + P_i(k)[P_i(k) + P_j(k)]^{-1} x_j(k) \tag{13}$$

The global error covariance is:

$$P_{\text{STF}}(k) = P_i(k)[P_i(k) + P_j(k)]^{-1} P_j(k) \tag{14}$$

where $x_i(k)$ is the local state estimation and $P_i(k)$ is the local error covariance of the track from sensor i at time k.

6   Experimental Results and Analysis

In order to facilitate the problem discussion, this paper supposes all the state estimations sent into the fusion center are in the same coordinate system, all the

sensors sample synchronously, and the delay time of data transmission is 0. The simulation designs four sensors to observe five targets at the same time. The sampling interval is l second, and the target-tracking lasts 200 seconds. The target tracks are shown in Fig. 1, and the observational data of the sensors are shown in Fig. 2.
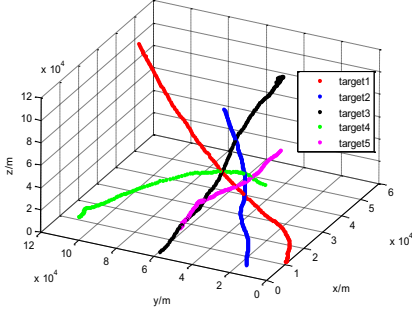
Fig. 1. The target tracks   **Fig. 2.** The observational data
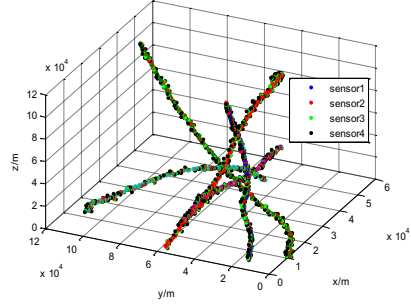
The comparisons between fusing "many" sensor tracks and fusing "all" sensor tracks on error in *x*-axis, *y*-axis and *z*-axis of target1 are shown in Fig. 3. We can see by fusing "many" sensor tracks, the error of system track is reduced greatly, which verifies fusing "many" sensor tracks instead of "all" sensor tracks can acquire the system tracks with higher accuracy. For other targets, there are similar conclusions.

(a) In *x*-axis   (b) In *y*-axis   (c) In *z*-axis
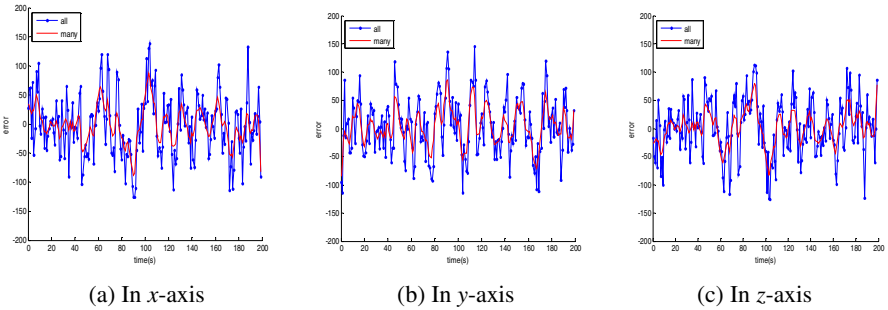
Fig. 3. The comparisons between fusing "many" sensor tracks and fusing "all" sensor tracks on error in *x*-axis, *y*-axis and *z*-axis of target1

The comparisons among MF, SF, WCF and STF on error covariance at the inflection points in *x*-axis, *y*-axis and *z*-axis of target1 are exhibited in Fig. 4. We can see STF has highest accuracy at the inflection points.

(a) In *x*-axis   (b) In *y*-axis   (c) In *z*-axis
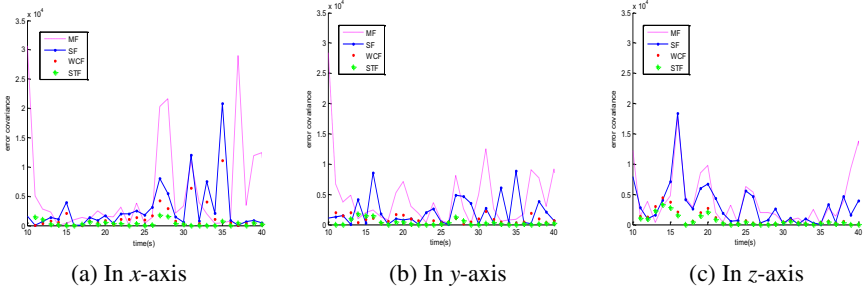
Fig. 4. The comparisons among MF, SF, WCF and STF on error covariance at inflection points in *x*-axis, *y*-axis and *z*-axis of target1

The average errors in x-axis, y-axis and z-axis of each target gained by MF, SF, WCF and STF are exhibited in Table 1. And Table 2 shows the average time for fusion of all targets by MF, SF, WCF and STF. From the two tables, we can see that MF and SF have a fast speed, but a low accuracy; WCF possesses of the highest accuracy, but with a very heavy calculation burden. The above three algorithms exist serious imbalance between accuracy and computational burden. However, STF almost approaches the accuracy of WCF with much less computational cost. The success of STF may lie in that it can significantly exclude those "bad" sensor tracks, which both reduces the fusion time and improves the fusion accuracy.

Table 1. The Comparison on Error

Algorithm	Target1			Target2			Target3			Target4			Target5		
	x	*y*	*z*	*x*	*y*	*z*	*x*	*y*	*z*	*x*	*y*	*z*	*x*	*y*	*z*
MF	60	64	67	29	58	61	55	36	33	56	57	53	45	48	47
SF	65	59	68	26	45	56	51	38	29	53	49	49	41	51	49
WCF	34	37	37	18	29	36	35	22	20	35	31	27	30	29	32
STF	36	42	41	24	35	39	31	30	26	32	36	35	31	33	38

Table 2. The Comparison on Fusion Time

Algorithm	Fusion Time (s)
MF	183
SF	225
WCF	525
STF	284

7   Conclusion

At present, most track fusion algorithms take into account the adaptation and the completeness of the fusion strategies, with little thinking over the quality of the information provided by the sensors. Different from them, the paper introduces a selective track fusion algorithm in the multi-sensor and multi-target environment. STF

has the following features: it eliminates the outlets by orthogonal polynomial regression, exploits the heuristics to guide track fusion, and fuse "many" instead of "all" sensor tracks, which not only facilitates the process of track fusion, but also improves the accuracy of the system tracks.

References

1. Willett, P.K.: The workshop on estimation, tracking and fusion: a tribute to Yaakov Bar-Shalom. Aerospace and Electronic Systems Magazine 17(3), 28–33 (2002)
2. Bar-Shalom, Y.: On the Sequential Track Correlation Algorithm in a Multisensor Data Fusion System. IEEE Transactions on Aerospace and Electronic Systems 44(1), 396–396 (2008)
3. Beugnon, C., Singh, T., Liaas, J., Saha, R.K.: Adaptive track fusion in a multisensor environment. In: Proc. of the ISIF, pp. 24–31. IEEE Press, Paris (2000)
4. Li, H., Cheng, K., Zhang, A., Shen, Y.: Adaptive Algorithm for Multisensor Track Fusion with Feedback Architecture. Chinese Journal of Computers 29(12), 2232–2237 (2006)
5. Chen, H.M., Bar-Shalom, Y.: Track association and fusion with heterogeneous local trackers. In: Proc. of the 46th IEEE Conf. on Decision & Control, pp. 2675–2680. IEEE Press, New Orlean (2007)
6. Watson, G.A., Rice, T.R., Alouani, A.T.: An IMM architecture for track fusion. In: Signal Proc. of SPIE Acquisition, Tracking, and Pointing, Orlando, FL, vol. 4052, pp. 2–13 (2000)
7. Chang, K., Chong, C.Y., Mori, S.: Analytical and Computational Evaluation of Scalable Distributed Fusion Algorithms. IEEE Transactions on Aerospace and Electronic Systems 46(4), 2022–2034 (2010)
8. Yuan, Q., Dong, C.Y., Wang, Q.: An adaptive fusion algorithm based on ANFIS for radar/infrared system. Expert Systems with Applications 36(1), 111–120 (2009)
9. Duan, Z.S., Li, X.R.: Lossless Linear Transformation of Sensor Data for Distributed Estimation Fusion. IEEE Transactions on Signal Processing 59(1), 362–372 (2011)
10. Hu, Y.Y., Duan, Z.S., Zhou, D.H.: Estimation Fusion with General Asynchronous Multi-Rate Sensors. IEEE Transactions on Aerospace and Electronic Systems 46(4), 2090–2102 (2010)

The Bystander Effect: Agent-Based Simulation of People's Reaction to Norm Violation

Charlotte Gerritsen

Vrije Universiteit Amsterdam, Department of Artificial Intelligence
De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
`cg@few.vu.nl`
Netherlands Institute for the Study of Crime and Law Enforcement
De Boelelaan 1077a, 1081 HV Amsterdam, The Netherlands
`CGerritsen@nscr.nl`

Abstract. The bystander effect is a well-known phenomenon in criminology, stating that bystanders tend to inhibit people's tendency to intervene in situations where norms are violated. This paper presents an agent-based simulation model of this phenomenon. The simulation model presented demonstrates the decision process of an agent for norm violation situations with different characteristics, such as high versus low personal implications. The model has been tested by performing a number of case studies. The outcome of these case studies show that the model is able to represent the behaviour of bystanders as expected based on various experimental studies.

Keywords: Bystander effect, Psychology, Cognitive modelling, Agent-based Simulation, Norm violation.

1   Introduction

In 1964 a young woman, named Kitty Genovese, was stabbed to death right outside of the apartment building in which she lived [1]. A newspaper claimed that 38 neighbours witnessed the event for half an hour but did not intervene. This claim was later proven inaccurate (no one could actually observe the entire assault), but the fact that a woman was murdered while bystanders observed part of the attacks (screaming, people fighting) without intervening is shocking. This case has been the starting point of research into the phenomenon of (non-)intervention (or the so-called bystander effect).

The main goal of the research presented in this paper is to develop a simulation model of the effect bystanders can have on people's reaction to a situation that (potentially) needs intervention e.g. because of norm violation or in case of an emergency.

The results of this study can be used in different domains. In the field of criminology, it may be useful to gain more insight in the mechanism of the bystander effect. This can help answering questions like: When do bystanders intervene? and What factors can help increase the number of interventions?

In the area of artificial intelligence this research can be helpful to create more realistic agents. These agents can be used for training purposes (e.g. for police agents) in a simulated setting, but can also be interesting in virtual societies (e.g. second life).

In this paper, the theory of the bystander effect will be explained in more detail in Section 2. In Section 3 an overview of related work is shown to demonstrate the innovative aspects of this study. The modelling approach and the simulation model are explained in Sections 4 and 5 and Section 6 shows a case study to illustrate the model. The paper is concluded with a discussion section (Section 7).

2   Bystander Effect

Bystander effect is a term used in a situation when people base the decision of acting in a certain situation (e.g. a norm violating situation or an emergency situation) on their bystanders. Imagine being in a situation in which intervening might be necessary e.g. you see someone falling on the ground. This could be caused by a heart attack, but maybe the person just tripped over his shoelace. Do you decide to help or not?

Research by [2, 3] demonstrated that the presence of other people decreases our willingness to help a victim in an emergency situation. Later studies, which are described in the next section, demonstrated that this is not only the case in emergency situations, but also for example in case of norm violations.

The main problem seems to be that people do not know how they should act in a given situation. They are not unwilling to help but do not know if their interpretation of the situation is correct. They seek for confirmation by looking at the behaviour of other bystanders, and let social cognition have a large influence on their behaviour.

Latané and Nida [4] distinguish three social psychological processes that might occur when an individual is in the presence of other people to explain their social inhibition of helping namely *audience inhibition* (you run the risk of embarrassment if you decide to intervene while you misinterpreted the situation), *social influence* (you confirm to the behaviour of others) and *diffusion of responsibility* (reduction of the psychological costs of non-intervention. Why should you help while all the other bystanders are also capable but do not do a thing?)

3   Related Work

In Section 3.1 an overview is given of related work in the area of social psychology. Similarly, in section 3.2 an overview of related work in the area of agent-based simulation is provided.

3.1   Social Psychology

In the field of social psychology a number of studies have been performed to investigate the effect of bystanders on behaviour. As mentioned in the previous section [2, 3] demonstrated that the presence of other people decreases our willingness to help a victim in an emergency situation.

Later studies demonstrated that the presence of others not only decreases our willingness to help in an emergency situation but that people are also less likely to answer the door [5], or leave a large tip in a restaurant [6] in the presence of others.

Chekroun and Brauer [7] conducted field studies to explore the influence of the number of bystander-observers on the likelihood of social control. The results of their

studies make clear that perceived personal implication moderates the extent to which people are inhibited by the presence of others when they decide whether they should exert social control or not.

The difference between the current research and the work mentioned above is the approach. The researchers of the mentioned articles conduct field studies to investigate the effect of a certain action on the behaviour of bystanders, with the number of bystanders as dependent variables. In the research presented in this paper, methods from the area of agent-based simulation are used to analyse the decision-making process, on a cognitive level. No real life experiments have been performed, but a model has been developed to simulate behaviour, given different parameters.

3.2   Agent-Based Simulation

In the last decades, there has been a growing interest in the area of Agent-Based Social Simulation (ABSS). In ABSS, which integrates approaches from agent-based computing, computer simulation, and the social sciences, researchers try to exploit agent-based simulation to gain a deeper understanding of social phenomena [8].

This approach has been used in a large number of studies, some of which address the criminological domain. For example in [9] an agent-based modelling approach has been presented for decision making, which integrates rational reasoning based on means-end analysis with personal psychological and biological aspects. In this project the authors use the same approach as presented in the current paper, namely simulating a decision process. However, the domain that they investigate is the domain of decision making in the case of street robbery while the work presented here focuses on the effect of bystanders on intervention.

Other ABSS studies in the field of criminology are the work by [10], which presents a model of deterrence with emphasis on the social network and the perceived sanctions. The paper leaves the mental aspects unaddressed, while the current paper also focuses on the cognitive aspect of the decision making process.

In [11], the authors model the process of emotion contagion. The focus is on cognitive aspects as is the focus of the model presented in the current paper. However, the authors do not apply their model to a real life situation.

With respect to norm violating behaviour many formal approaches exist. In [12] and [13] approaches for representing norm-governed multi-agent systems are introduced. More specific for norm violation, in [14] procedural norms are discussed that can for example be used to motivate agents that play a role in recognizing violations or applying sanctions. In [15] a formalism is introduced to detect, sanction and repair violations of norms. The difference between these papers and the research presented in the current paper is that we focus specifically on norms in a human situation instead of norms in software agents.

4   Modelling Approach

To model the various relevant aspects of the bystander effect in an integrated manner poses some challenges. On the one hand, qualitative aspects have to be addressed, such as desires, and intentions, and some aspects of the environment such as the

observation that an action occurs and that the other bystanders do not intervene. On the other hand, quantitative aspects have to be addressed, such as the number of bystanders and the level of seriousness of the norm violation.

The modelling approach based on the modelling language LEADSTO [16] fulfils these requirements. It integrates qualitative, logical aspects and quantitative, numerical aspects. This integration allows the modeller to exploit both logical and numerical methods for analysis and simulation. LEADSTO enables to model direct temporal dependencies between two state properties in successive states by means of *executable dynamic properties*, which are comparable to rules as occurring in specifications of a simulation model; for example:

If	*in the current state, state property p holds,*
then	*in the next state, state property q holds*

Here, atomic state properties can have a qualitative, logical format, such as an expression desire(d), expressing that desire d occurs, or a quantitative, numerical format such as an expression   belief(norm,y) which expresses that y is a threshold above which actions are considered norm violations. For more details of the language LEADSTO, see [16]. Based on LEADSTO, a dedicated piece of software has been developed [16]. The LEADSTO Simulation Environment takes a specification of executable dynamic properties as input, and uses this to generate simulation traces.

5   Simulation Model

The decision model for a single agent consists of a set of LEADSTO rules. The model is depicted in Figure 1. The model of the decision making process shown in Figure 1 is inspired by the so-called BDI-model [17]. The BDI-model bases the preparation and performing of actions on beliefs, desires and intentions. It incorporates a pattern of reasoning to explain behaviour in a refined form. Instead of a process from desire to action in one step, as an intermediate stage first an intention is generated. An action is performed when the subject has the intention to do this action and it has the belief that the opportunity to do the action is there. The BDI model is extended by introducing rules that formalise the theory by [4]. The model consists of rules to determine the belief that the agent is personally responsible, to determine the desire to help, the belief that there is an opportunity to help and eventually to perform the action. These sets of rules are explained below.

Personal responsibility

LP1 "If you observe that others do not intervene this leads to the belief that intervention will be evaluated negatively (*social influence*)":
  observe(no_intervention_by_others) →belief(intervention_will_be_evaluated_negatively)

LP2 "If you observe a number of bystanders present then you will believe that the bystanders can observe you":
  ∀n:INTEGER
  observe(number_of_bystanders(n)) → belief(others_observe_me(n))

LP3 "If you observe an action and you believe that others can observe you and that intervention will be evaluated negatively this will lead to the belief of *audience inhibition* with value n":

∀a:ACTION ∀n:INTEGER
observe(a) ∧ belief(others_observe_me(n)) ∧ belief(intervention_will_be_evaluated_negatively)
→ belief(audience_inhibition(n))

LP4 "The number of bystanders that you observe determines your belief about the costs of intervention. The higher the number of bystanders the higher the costs (*diffusion of responsibility*)":

∀n:INTEGER
observe(number_of_bystanders(n)) → belief(costs_intervention(n))

LP5 "Your belief about the costs of intervention combined with your belief of audience inhibition determines your belief on personal responsibility. The lower the costs and audience inhibition the higher the belief of personal responsibility":

∀n1, n2:REAL
belief(audience_inhibition(n2)) ∧ belief(costs_intervention(n1)) ∧ n1<thn1 ∧ n2<thn2 →
belief(personal_responsibility)

The thresholds used in LP5 (thn1 and thn2) are linked to the level of seriousness. If you believe that a certain violation is very serious, and thus feel highly personally implicated then you will probably less likely let the costs of intervention or audience inhibition stop you from intervening. The threshold for non-intervention is high. However, if you believe a violation is not serious at all, then you will let the costs and audience inhibition keep you from intervening. In this case the threshold for non-intervention will be low.

Desire to help

LP6 "When you observe an action and you believe that this action has a seriousness of value s and you see that there are bystanders present and that these bystanders do not intervene then this leads to the belief that the level of seriousness of the action is s divided by the amount of bystanders times α. Here, α is a parameter that determines the influence of the group; α is by default set to 0.5. You adjust your opinion on the level of seriousness based on the non intervention of the others":

∀s:REAL ∀a:ACTION ∀n:INTEGER
observe(a) ∧ observe(no_intervention_by_others) ∧ observe(number_of_bystanders(n)) ∧
belief(has_seriousness(a,s)) → belief(seriousness, s/n^α)

LP7 "If you belief that the action has a level of seriousness of x and you belief that a certain norm has value y and the action violates the norm (x>y) then you believe that there is an emergency":

∀x,y:REAL
belief(seriousness, x) ∧ belief(norm,y) ∧ x>y → belief(emergency)

LP8 "If you believe that there is an emergency then you have the desire to help":

belief(emergency) →desire(i)

Intention to help

LP9 "When you believe that you are personally responsible to help, and you have the desire to help then you have the intention to help":

desire(i) ∧ belief(personal_responsibility) → intention(i)

Opportunity to help

LP10 "If you belief that you are capable to help and have the required resources then you believe that you have the opportunity to help":

belief(capable(i)) ∧ belief(resources(i)) → belief(opportunity_for(i))

Performance of Action

LP11 "The intention to help combined with the belief that you have the opportunity to help leads to the actual intervention":

intention(i) ∧ belief(opportunity_for(i)) → perform(i)
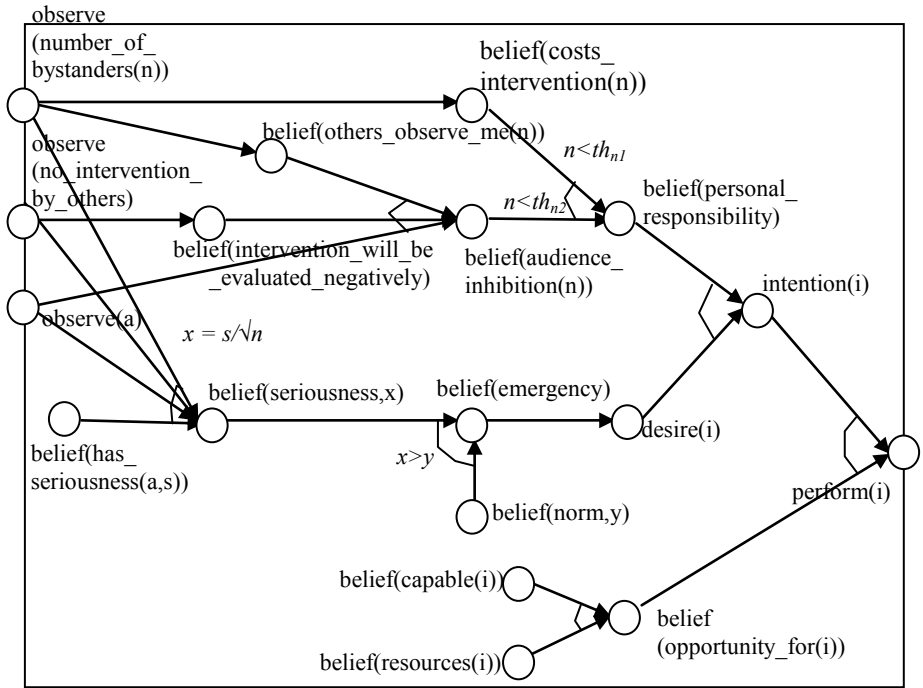
Fig. 1. Simulation model

6   Case Studies

To illustrate the behaviour of the model we present two case studies: one case study with a norm violation in a situation with high personal implication and one case study with a norm violation in a situation with low personal implication. Recall that personal implication indicates the level of responsibility a person feels for a particular situation: when you feel it is your duty to intervene in a certain situation you feel highly personally implicated (e.g. when someone damages something in your home). However, when you do not feel any responsibility to intervene, you have a low

personal implication (e.g. when you see someone shoplifting). The norm violating situations used in the case studies are based on the experiments presented in [7].

In Section 6.1 the case studies will be described and the results of the simulations are shown in Section 6.2.

6.1   Description

Low personal implication

Chekroun and Brauer [7] conclude that whether or not people feel personally implicated, has a great impact on their incentive to intervene. According to their research people feel low personal implication in case of a norm violation in an elevator in a shopping mall. This is caused by the fact that the shopping mall belongs to a corporate business and professional cleaning personnel is responsible for keeping the shopping mall clean.

In their experiment [7], a young man is waiting for the elevator. When people join him, he enters the elevator with them. Once the doors are closed, he takes out a marker and draws something on the wall. He gives the other people in the elevator the opportunity to react, before leaving the elevator at the first floor. The results of the experiment show that the number of bystanders has an effect on the intervention rate. The more bystanders were present, the lower the probability that someone expressed his or her disapproval.

High personal implication

People feel highly personally implicated in situations in which they feel it is their personal obligation to intervene. Chekroun and Brauer [7] use a park as a situation in which people feel highly personally implicated. The maintenance of the park is paid via local taxes and thus by the inhabitants. Further, aspects with respect to rising consciousness about the environment play an important role in feeling personally implicated.

To test the intervention behaviour and the effect of bystanders, the following experiment was performed in [7]. Two females are walking through a park. As soon as they approach other people, they start to drink from a plastic bottle and throw the bottle in the bushes. They make sure that all bystanders notice this. Then they walk away, and give the bystanders the opportunity to react to the littering.

The results of this experiment show that in this case, the number of people present does not affect the intervention behaviour of the bystanders.

6.2   Example Simulation Traces

We have used the situations described in Section 6.1 as scenarios to demonstrate our model. For each of the situations (low personal implication and high personal implication), we have run the simulation model with various parameter settings for the amount of bystanders, the value of the norm and the level of seriousness.

Table 1 shows the parameter settings (first four columns) and results (last four columns) of some of these runs. The first column shows how many bystanders are present. The second column indicates the agent's behavioural norm. A norm with

value 0.1 means that the agent believes very quickly that norms are violated; while a value of 0.8 means that the agent will not believe this very quickly. The seriousness (column 3) determines how serious the agent believes a certain action usually is (0.1 means that an action is not considered to be very serious, while an action rated with 0.9 is considered to be very serious). The value for the thresholds used in LP5 (thn1 and thn2, which are currently taken equal) is mentioned in the fourth column. Based on these settings, the fifth column shows the updated seriousness (which is the result of the calculation performed by LP6), i.e., the agent's perceived seriousness of an action in one particular situation. The final three columns show whether or not the agent feels responsible to intervene, has a desire to intervene and performs an intervention.

Table 1. Results simulation runs

#by-standers	norm	serious-ness	threshold	updated seriousness	personal responsibility	desire	inter-vention
1	0.1	0.2	2	0.20	yes	yes	yes
3	0.1	0.2	2	0.18	no	yes	no
4	0.1	0.2	2	0.17	no	yes	no
5	0.1	0.2	2	0.17	no	yes	no
1	0.1	0.5	5	0.50	yes	yes	yes
3	0.1	0.5	5	0.45	yes	yes	yes
4	0.1	0.5	5	0.44	yes	yes	yes
5	0.1	0.5	5	0.43	no	yes	yes
1	0.1	0.9	9	0.90	yes	yes	no
3	0.1	0.9	9	0.81	yes	yes	yes
4	0.1	0.9	9	0.78	yes	yes	yes
5	0.1	0.9	9	0.77	yes	yes	yes
1	0.5	0.1	1	0.10	no	no	yes
3	0.5	0.1	1	0.09	no	no	no
4	0.5	0.1	1	0.09	no	no	no
5	0.5	0.1	1	0.09	no	no	no
1	0.5	0.5	5	0.50	yes	no	no
1	0.5	0.6	6	0.60	yes	yes	no
1	0.5	0.9	9	0.90	yes	yes	yes
3	0.5	0.9	9	0.81	yes	yes	yes
4	0.5	0.9	9	0.78	yes	yes	yes
5	0.5	0.9	9	0.77	yes	yes	yes
1	0.8	0.9	9	0.90	yes	yes	yes
2	0.8	0.9	9	0.84	yes	yes	yes
3	0.8	0.9	9	0.81	yes	yes	yes
4	0.8	0.9	9	0.78	yes	no	no

These results show that the model is able to reproduce the behaviour as found in the experiments performed by [7]. Indeed, intervention depends on the amount of bystanders and the personal commitment. The number of bystanders is important in the low personal implication scenarios (seriousness 0.1-0.5), while the bystanders do not have a large effect on the intervention behaviour in the high personal implication scenarios (seriousness 0.6-0.9).

7   Discussion

In this paper we have developed a simulation model to demonstrate the behaviour of bystanders in norm violating situations. This model was tested in two case studies

with multiple different scenarios. In these case studies we used situations with low and high personal implication and tested the model with different settings for the number of bystanders, the level of the norm, seriousness of the violation and thresholds. The results of the simulations show that the model can replicate the behaviour of the agent as expected based on the results of various real life experiments [7].

The model can be useful in different domains. In the field of criminology the model can support researchers to gain more insight in the decision process behind the bystander effect. For instance: Under what circumstances will people intervene? Do they intervene when they see someone stealing a bike? Or when someone gets murdered like in the Kitty Genovese case?

Within the domain of Artificial Intelligence, the results are interesting for the development of more realistic virtual agents. These agents play an important role, e.g., in training simulations for police officers. Thus, the model can be used as a first step in developing 'virtual bystanders' for such applications.

Finally, the results can also be helpful for policy makers. For instance, it may provide ideas about how to change environmental design in such a way that bystander interventions are encouraged. This might make it possible to reduce the amount of police effort, and thus to reduce costs.

This paper is a starting point of agent-based simulation in the area of the bystander effect. Future work will include research into other relevant aspects (e.g., environmental aspects of the location of the violation, appearance of the offender), and to a more extensive validation of the model.

References

1. http://en.wikipedia.org/wiki/Murder_of_Kitty_Genovese
2. Latané, B., Darley, J.M.: Group inhibition of bystander intervention in emergencies. Journal of Personality and Social Psychology 10, 215–221 (1968)
3. Latané, B., Darley, J.M.: The unresponsive bystander: Why doesn't he help? Appleton-Century-Crofts, New York (1970)
4. Latané, B., Nida, S.: Ten years of research on group size and helping. Psychological Bulletin 89, 308–324 (1981)
5. Levy, P., Lundgren, D., Ansel, M., Fell, D., Fink, B., McGrath, J.E.: Bystander effect in a demand-without-threat situation. Journal of Personality and Social Psychology 24, 166–171 (1972)
6. Freeman, S., Walker, M., Bordon, R., Latané, B.: Diffusion of responsibility and restaurant tipping: Cheaper by the bunch. Personality and Social Psychology Bulletin 1, 584–587 (1975)
7. Chekroun, P., Brauer, M.: The bystander effect and social control behaviour: the effect of the presence of others on people's reactions to norm violations. European Journal of Social Psychology 32, 853–867 (2002)
8. Davidsson, P.: Agent Based Social Simulation: A Computer Science View. Journal of Artificial Societies and Social Simulation 5(1) (2002)
9. Bosse, T., Gerritsen, C., Treur, J.: Combining Rational and Biological Factors in Virtual Agent Decision Making. Journal of Applied Intelligence 34(1), 87–101 (2004)

10. van Baal, P.H.M.: Computer Simulations of Criminal Deterrence: from Public Policy to Local Interaction to Individual Behaviour. Ph.D. Thesis, Erasmus University Rotterdam. Boom Juridische Uitgevers (2004)
11. Bosse, T., Pontier, M.A., Treur, J.: A Computational Model based on Gross' Emotion Regulation Theory. Cognitive Systems Research Journal 11, 211–230 (2010)
12. Rubino, R., Omicini, A., Denti, E.: Computational Institutions for Modelling Norm-Regulated MAS: An Approach Based on Coordination Artifacts. In: Boissier, O., Padget, J., Dignum, V., Lindemann, G., Matson, E., Ossowski, S., Sichman, J.S., Vázquez-Salceda, J. (eds.) ANIREM 2005 and OOOP 2005. LNCS (LNAI), vol. 3913, pp. 127–141. Springer, Heidelberg (2006)
13. Esteva, M., Padget, J., Sierra, C.: Formalizing a Language for Institutions and Norms. In: Meyer, J.-J.C., Tambe, M. (eds.) ATAL 2001. LNCS (LNAI), vol. 2333, pp. 348–366. Springer, Heidelberg (2002)
14. Boella, G., van der Torre, L.: Substantive and Procedural Norms in Normative Multiagent Systems. Journal of Applied Logic 6(2), 152–171 (2008)
15. Vazquez-Salceda, J., Aldewereld, H., Dignum, F.: Implementing Norms in Multiagent Systems. In: Lindemann, G., Denzinger, J., Timm, I.J., Unland, R. (eds.) MATES 2004. LNCS (LNAI), vol. 3187, pp. 313–327. Springer, Heidelberg (2004)
16. Bosse, T., Jonker, C.M., van der Meij, L., Treur, J.: A Language and Environment for Analysis of Dynamics by Simulation. International Journal of Artificial Intelligence Tools 16(3), 435–464 (2007)
17. Rao, A.S., Georgeff, M.P.: Modelling Rational Agents within a BDI-architecture. In: Allen, J., et al. (eds.) Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning (KR 1991), pp. 473–484. Morgan Kaufmann (1991)

Multi Agent Carbon Trading Incorporating Human Traits and Game Theory

Long Tang, Madhu Chetty, and Suryani Lim

Gippsland School of Information Technology, Monash University, VIC-3842, Australia
Long.Tang@aiviet.com, Madhu.Chetty@monash.edu, Suryani.Limmy@gmail.com

Abstract. Carbon trading scheme is being established around the world as an instrument in reducing global GHG emission. Being an emerging market, there only a few simple simulation studies related to carbon trading that have been reported. In this paper, we propose a novel carbon trading simulator capable of modeling traits of human traders in carbon markets. The model is driven by the concept of Nash equilibrium within an agent based modeling paradigm. The model is capable of implementing crucial issues such as carbon emissions, Marginal Abatement Cost Curve (MAC), and complex human trading behaviour. Experiments carried out provide insights into interaction between traders' behaviour and how the interaction affects profitability.

Keywords: Carbon emission trading, game theory, multi agent system, marginal abatement curve, Nash equilibrium.

1 Introduction

Carbon trading is a term commonly used to reflect the activity of trading carbon credits (GHG credits or units). A unit is a measure of allowable CO_2 emission, so to be "greener" is to reduce GHG emissions. Carbon markets, based on market competition, spread the financial cost of carbon emissions reduction across different nations.

Being an emerging domain of research, limited literature is available on modeling of carbon trading. A group of researchers on carbon trading markets have restricted their research to using human participants [1, 16]. Others have used computer models [9, 10, 14, 15]. Mizuta and Yamagata were probably the first to present carbon trading simulation using agent modeling approach [10]. They developed a generic framework called Artificial Society with Interacting Agents (ASIA). In the ASIA framework, an agent represents a country. Their simulations used 12 countries (6 Annex I countries and 6 Non Annex I countries) covering year 2008 to 2012, and the reduction is distributed evenly across the trading years. They found that an agent's decision to act on GHG reduction early or late was influenced by the technology advancements and other (unspecified) factors.

Mizuta and Yamagata extended ASIA to also include human players [11, 12], and made three important observations. Firstly, that human players made dishonest and irrational bids resulting in no deals. To deal with this problem, they

restricted human players to Double Auction trading. Secondly, human sellers controlled price at the expense of human buyers. Finally, when human players could invest in technological advancement for self-cutting, they invested more heavily in technology and price control became unpopular. However, it was unclear how this change in behaviour affected the selling price, and therefore, profits. To answer this question, we propose a framework in which agents tend to control selling price (having the trait of "greed"), less likely to control selling price (having the trait of "goodwill") and agents who have the same likelihood to have the trait of greed or goodwill ("neutral"). Our proposed framework is described in the next section.

2   The Proposed Model

As mentioned in the previous section, research in carbon markets show that human players exhibit the "greed" and "goodwill" characteristics. Thus, the carbon trading markets become very similar to Prisoners' Dilemma, the classic problem in game theory that explains why two people do not cooperate even if it is in their best interest to do so.

To get a clearer understanding of our trading model, we elaborate the Prisoners Dilemma as follows. Let us consider two persons who have been arrested for a crime. Now, if one person confesses to implicate the partner and the partner does not also confess, then the person who made the confession goes free and the partner gets ten years of imprisonment. If both of them confess, then they each get five years. If neither of them confess, then each of them gets two years. The best outcome for an **individual** criminal is to go free, and the worst outcome is to be in prison for ten years. But, the best outcome for **both** criminals is when they cooperate: none of them confess and each gets only two years. However, a "selfish" criminal is more likely to confess since there is a chance to be free. In this case, a "greedy" trader is similar to the "selfish" criminal who is more likely to confess, while a "goodwilled" trader is less likely to confess.

In our framework, we propose three major categories of traders: Profit Driven (PD) to represent "greed", Green Driven (GD) to represent "goodwill", and Neutral who switches between PD and GD. Note that the Neutral behaviour was not a distinct behaviour observed in [11, 12], but we would like to evaluate the performance of such a flexible trader. A PD trader attempts to maximize monetary gain at the expense of other traders. The classic example of PD motivation is "buy low and sell high". On the other hand, a GD trader makes trading decisions with monetary gain of secondary importance; it focuses on having deals approved. However, this does not mean that a GD trader does not seek to make a gain. Rather, it means that a GD trader is more likely to seek a lower profit margin than PD one. In other words, in response to the same market situation, a GD trader tends to bid higher and ask lower than a PD one.

The next sections explain how to implement these human traits in computer agents, how to use game theory in making decisions, and how the trading mechanism works.

2.1   Coding Trader Traits in Computer Agents

The complex behaviour of traders can be implemented using the concept of payoff matrix of game theory by changing the values of the matrix. The payoff values (u) are in the integer range of $[0...10]$.

The three columns in Figure 1 are actually the three payoff matrices and represent the three different distinct agent personalities: PD (profiteering or ugly), Neutral and GD (environmental trait or good). For a PD agent, the gain will be made at the expense of her trading partner. A GD agent, on the other hand, seeks co-operation above everything else. A neutral agent has the equal probability of being a PD or GD. The next section now discusses the proposed two-stage trading mechanism, and how the payoff matrices affect the behaviour of the agents.

Profiteering			Neutral			Environmental		
Pr(Seller/Buyer is PD)=1 Pr(Seller/Buyer is GD)=0			Pr(Seller/Buyer is PD)=0.5 Pr(Seller/Buyer is GD)=0.5			Pr(Seller/Buyer is PD)=0 Pr(Seller/Buyer is GD)=1		
	PD Buyer	GD Buyer		PD Buyer	GD Buyer		PD Buyer	GD Buyer
PD Seller	2 2	9 1	PD Seller	2 3	6 3	PD Seller	1 2	7 1
GD Seller	1 9	3 3	GD Seller	2 8	5 5	GD Seller	3 7	8 8

Fig. 1. The payoff matrices representing PD (profiteering), Neutral and (GD) environmental

2.2   The Two Stage Trading Algorithm

In making trading decisions, a nation agent in the model relies on four pieces of information: emissions profile, Marginal Abatement Cost (MAC) profile, market information, and payoff matrices. Each piece of information provides the agent with a view of the overall situation from different angles.

Based on the emission profile, an agent can answer how polluted or green it is, and decide what its emissions target is in a certain year. Using the MAC profile, an agent knows how much it costs to reduce its carbon emission domestically. With the market information, an agent knows what the global price of each unit of carbon credit is. Using the payoff matrix, an agent decides on the "optimal" course of action. Together, these views give a big picture of the market and hence enable an agent to form conjectures about trading decision.

There are two distinct stages in an agent's decision making process and the output of the first stage forms the input of the second stage (see Figure 2). Stage 1 determines the carbon credit quantity for selling (ask) or buying (bid) and the preliminary selling or buying price P_1 using only the agents MAC and market unit price. The advertised price P_2 in stage 2 is the result of a mark-up or mark-down of preliminary price P_1 produced in stage 1. The next section describes the algorithm in stage 1.
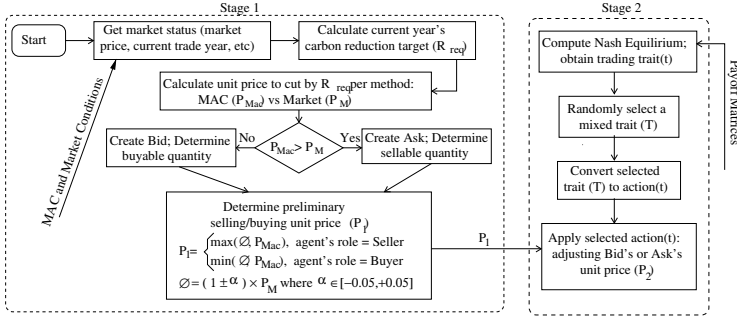
Fig. 2. The proposed two stage trading algorithm

Stage One. Every agent in the model has a reduction target (R_{req}), which is spread evenly across the trading years. To achieve this target, an agent can either self reduce, sell or buy carbon credits. Let P_{Mac} be an agent's MAC price to cut emissions by R_{req}, so $P_{Mac} = MAC(R_{req})$, and P_M the market price of R_{req}.

To obtain an optimal decision, an agent examines its MAC and compares P_{Mac} with P_M for reducing R_{req}. Comparing P_{Mac} with P_M, there are three possible outcomes: P_{Mac} is less than, equal to, or greater than P_M. When P_{Mac} equals to P_M, an agent has no carbon credits to sell nor buy. When $P_{Mac} \neq P_M$, the agent has to decide whether to buy or sell.

Let P_{M1} be the market price in case P_M is greater than P_{Mac}. Similarly, let P_{M2} be the market price in case P_M is less than P_{Mac}. Figure 3 shows the mechanism of an agent determining the sellable/buyable quantity and the corresponding unit price by comparing P_{Mac} and P_M.
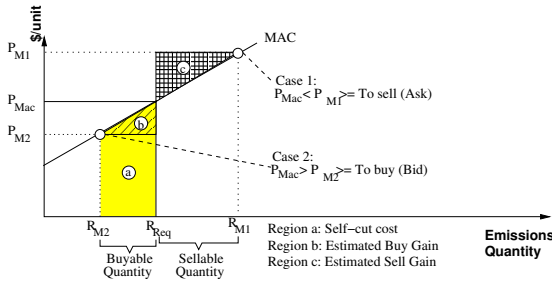
Fig. 3. An agent compares P_{Mac} with P_M to determine sellable or buyable quantity and the corresponding unit price

Let $R_\triangle$ be the trading quantity that an agent would propose to make a trading gain:

Case 1: Sellable quantity $R_\triangle = |R_{req} - R_{M1}|$
Case 2: Buyable quantity $R_\triangle = |R_{req} - R_{M2}|$

Let P_1 be the preliminary trading price of $R_\triangle$. In real world, traders would make offer with reference to the current market price P_M. Traders' offers could be either above, equal to, or below the market price. Each trader would propose a different price, so in stage 1 of the algorithm, the model has a price fluctuation function that randomly sets P_1 within $\pm 5\%$ difference of P_M. In addition, the fluctuation is constrained so that agents would not "sell below cost" (sell below P_{Mac}) or "buy at a loss" (buy above P_{Mac}). Thus, the definition of P_1 is as follows:

$$P_1 = \begin{cases} max(\oslash, P_{MAC}) & \text{Agent's role} = \text{Seller} \\ min(\oslash, P_{MAC}) & \text{Agent's role} = \text{Buyer} \end{cases}$$

$$\oslash = (1 \pm \alpha) \times P_M, \alpha \in [-0.05, +0.05]$$

Let $Cost_{Mac}$ be the cost (in dollar) to self-cut emissions by an amount of $\mid a - b \mid$. It is the area under MAC curve:

$$Cost_{Mac} = \int_a^b MAC$$

hyphenationreceiv-a-ble Trading money exchanged between buyers and sellers is referred to as *payable* (amount of money a buyer expected to buy) and *receivable* (amount of money a seller expected to receive). For $R_\triangle$ carbon credits traded at price P_1, payable/receivable is calculated as follows: $Payable_{R_\triangle}$ or $Receivable_{R_\triangle} = R_\triangle \times P_1$. With payable/receivable and $Cost_{Mac}$ known, an agent can estimate the potential gain. An agent gain from either selling or buying carbon credits. For $R_\triangle$ trading quantity:

- Sale gain, the amount of difference between the self-cut cost and the amount of payment received from buyer (see Figure 3).

$$Sale\ Gain = Receivable_{R_\triangle} - Cost_{Mac} = R_\triangle \times P_1 - \int_{R_{req}}^{R_{M_1}} MAC$$

- Buy gain, the amount of difference between the supposed self-cut cost and the actual cost of buying credits.

$$Buy\ Gain = Cost_{Mac} - Payable_{R_\triangle} = \int_{R_{M_2}}^{R_{req}} MAC - R_\triangle \times P_1$$

Once the gain has been calculated using the MAC function as above, we apply game theory using Nash equilibrium (NE) to adjust the offering or buying price, and the trader traits (either green or greedy) will affect how much the unit price will be adjusted. The price adjustment is performed in the second stage of the algorithm, and it is described next.

Stage Two. In the second stage of the two-stage trading algorithm, game theory is applied to select an "optimal" price for a given agent type (PD or GD).

A core component of a payoff matrix is the payoff function. Suppose that there are n Game Players in a game and x_i is the type of player i (GD, PD). Let x be a

type profile which is a set containing the type of each player $x = (x_1, x_2, \ldots, x_n)$. Payoff function of a player i takes a type profile x as input and returns payoff value as output:

$$u_i(x) \text{ where } i = \text{ player and } x = \text{ type profile.}$$

In the model, a payoff value (or utility) has an integer scale from zero to ten (lowest to highest). A Game Player i specifies its preferred type x_i in type profile x. A payoff value is not only a preference but also a measure of the quality of outcome to a player i when selecting a type x_i in type profile x.

Let P_M be the current market price, MAC the marginal abatement cost curve of a trader, MAC^{-1} the inverse of MAC. Let R_M be the trader's potential trading quantity at market price: $R_M = MAC^{-1}(P_M)$. With P_M and R_M known, a real world trader has with three options:

1. Price adjustment to make his/her bid/ask being more or less competitive;
2. Quantity adjustment to make more or less profit; or
3. Combination of (1) and (2).

In our trading system, agents propose how much units to buy or sell, and when a deal is struck, the settled deal is the lower of the two. This is because the trading system enforces a partial order when there is a difference between "supply" and "demand" quantity. In stage-2 of the algorithm, agents apply price adjustment to P_1 reflecting its type. The logic behind this adjustment is as follows. Let β the adjustment percentage intended for P_1:

$$\beta \equiv f(t, o) \text{ where } t : type \in (PD, GD) \text{ and } o : role \in (Seller, Buyer)$$

Let P_2 be the advertised price in stage-2:

$$P_2 = P_1 \times (1 \pm \frac{u}{10} \times \beta) = P_1 \times (1 \pm \frac{u}{10} \times f(t, o))$$

where u: payoff amount $\in [0, 10]$.

The price adjustment function with GD type is defined as follows:

$$f(t = GD, o) = \begin{cases} random(-0.3, 0.1), & o = \text{ Seller} \\ random(-0.1, 0.3), & o = \text{ Buyer} \end{cases}$$

The random function used in the implementation of the model is a uniform distribution pseudo-random numbers generator [8]. When a GD agent decides to buy, the probabilities of positive ($P_2 \geq P_1$) and negative ($P_2 < P_1$) adjustments are as follows:

- **Case 1:** Probability of **positive** price adjustment

$$Pr(P_2 \geq P_1) = Pr(0 \leq \beta \leq 0.3) = 0.3 \times \frac{1}{0.3 - (-0.1)} = \frac{3}{4}$$

– **Case 2:** Probability of **negative** price adjustment

$$Pr(P_2 < P_1) = Pr(-0.1 \leq \beta < 0) = 1 - Pr(0 \leq \beta < 0.3) = 1 - \frac{3}{4} = \frac{1}{4}$$

As previously discussed, GD and PD types have opposite priorities. GD is more concerned about getting deals approved than maximizing trading gain whereas PD is aggressive in maximizing gain in every approved deal. In terms of pricing adjustment, a PD agent acts in the opposite manner to a GD agent. That is, a PD agent is likely to sell at a higher price and buy at a lower price as compared to a GD. Such a bias is captured in the following price adjustment function:

$$f(t = PD, o) = \begin{cases} random(-0.1, 0.3), & o = \text{Seller} \\ random(-0.3, 0.1), & o = \text{Buyer} \end{cases}$$

Payoff values are an agent's measure of quality of outcome and they are used to search for Nash equilibrium strategy in the payoff matrix. The higher the payoff value, the better the outcome is. Another role of the payoff value u is to act as the scaling factor of the price adjustment percentage β. The value of u indicates the strength of an agent's type; for example, two nations are of GD type, but a nation with $u = 9$ is more aggressive in getting the deal approved (by lowering its profit margin) than say a nation with $u = 7$.

3   Experiment Setup

Trading is conducted using First Price Sealed Bid auction (FPSB). Price is initially set at \$25/ton, based approximately on the recent trading prices reported at European Climate Exchange (ECX). The values of payoff matrices were given in Figure 1. We simulated six Annex 1 nations expected to lead the world in reducing GHG emissions. We excluded non Annex 1 nations mainly because there are no reliable MAC functions available. We err on the side of simulating fewer nations and use more realistic data. Since these six countries are expected to lead the world, their activities will probably dominate the carbon trading markets landscape, so the simulation results will still be meaningful. The six Annex 1 countries in the simulations are Australia, US, Russia, UK, Sweden, and Switzerland. Their MAC functions are linear curve fit of McKinsey's MAC estimates [2–7].

An experiment has 11 trading years, starting from year 2010 till 2020 (inclusive); a trading year has one trading round. All agents share a common reduction goal of 24% of the base year (1990) carbon emissions; 24% being the Australian reduction target conditional on adequate global agreement [13]. All agents must meet their reduction target by the end of a simulation either by self-cutting or trading.

The experiments were conducted under three trading scenarios: Neutral, Green Driven, and Profit Driven. A trading scenario is designed by aligning a majority of nation agents in the experiment with the similar trait. In the context of experiments, "majority" or "most" means the nation agents in the population are greater than 50%. For example, in Green Driven scenario, most agents are GD.

For each trading scenario, we setup an experiment and simulate it repeatedly ten times, so there are thirty simulations in total. The next section presents the results and analysis of results. Note that in the absence of any available literature devoted to similar type of investigations, it has not been possible to conduct comparisons with any other approaches reported earlier.

4   Results and Discussions

A snapshots of results in "Green Driven" trading scenario is given in Table 1. Similar to the findings of [12], nations tend to self cut in the first few years, and then self correct in the later years. Annotation 1 in Table 1 emphasizes the presence of a period (from year 2010 to 2013) when there is no trading activity. This special trading period is known as Cut-To-Grow (CTG): there is no incentive for that trader to buy carbon credits internationally. Rather, the trader would seek to sell credits and to self-cut carbon emissions at home because such actions would have a net positive effect on its economy [6].

Annotation 2 marks a rare event when nation agents in the model trade at a loss and such an event happens beyond the control of agents. Agents may experience loss when trading because of the rule of trading: partial bid is allowed and there is no reserve price or quantity.

Annotation 3 is an example of market price decline. From year 2014 to year 2020, we observed that market price decline happens less frequently than market price increase.

In Annotation 4, market price remains unchanged from 2016 to 2017. This situation arises because there is no successful deal in that period (Sale Gain = Buy Gain = 0). No deal happens from time to time when agents see no gain in trading (CTG for example); or proposed bids and asks simply do not match.

Table 1. A snapshot of simulation results under Green Driven trading scenario

Green Driven Trading Scenario

Run No	Year	Price ($/ton)	SaleGain ($)	ExtraSaleGain ($)	BuyGain ($)	ExtraBuyGain ($)	Run No	Year	Price ($/ton)	SaleGain ($)	ExtraSaleGain ($)	BuyGain ($)	ExtraBuyGain ($)	
1	2010	25	0	0	0	0	2	2010	25	0	0	0	0	*Annotation 2: Trading at a loss!*
	2011	25	0	0	0	0		2011	25	0	0	0	0	
	2012	25	0	0	0	0		...	...					
	2013	25	0	0	0	0		2018	32.82	−145.43	−290.86	5490.03	5452.59	
	2014	23.37	46748.4	45070.93	9582.17	910.91		2019	39.29	19146.82	18759.93	207131.35	179624.03	*Annotation 3: Market price decline*
	2015	25.51	48864.45	48477.56	26721.45	−785.87		2020	38.94	733733.33	698866.87	1380718.76	1031144.63	
	2016	30.82	256962.82	14076.01	739602.89	699244.06	3	2010	25	0	0	0	0	
	2017	31.86	23769.31	19861.04	144531.05	122353.97		...	...					
	2018	31.91	385636.02	−124632.45	3522297.33	3395835.46		2016	27.41	1157142.43	31813.98	287644.03	2613566.23	*Annotation 4: Unchanged market price*
	2019	32.04	64687.51	−23074.98	520837.01	498368.62		2017	27.41	0	0	0	0	
	2020	36.47	671660.18	641192.97	1114667.32	828540.48		...	...					

Annotation 1: Cut−to−Grow Period (CTG)

We now examines the relationship between trading trait and total trading gain (sale gain + buy gain). The yearly average total gain for three different scenarios are: (i) Profit Driven, (ii) Neutral, and (iii) Green Driven. For each trading scenario, the yearly total gains of simulations are averaged and displayed in Figure 4.

An agent's trading performance is measured by the amount of monetary gain it made through the trades. In Figure 4, the trading performances of three different
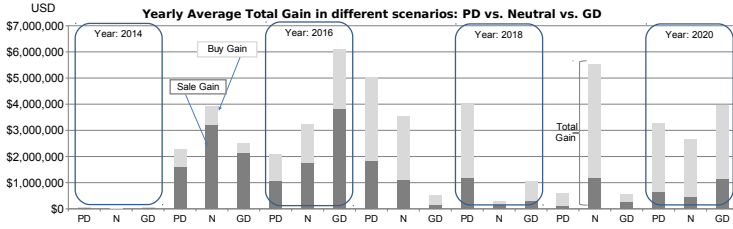
Fig. 4. Stacked-column chart shows Yearly Average Total Gain in different scenarios: PD vs. Neutral vs. GD

scenarios in each year are compared to each other. The higher the bar, the better the trading performance is. A trading scenario is said to have a high performance year (high year) when agents under the trading scenario make the most gain as compared to agents in other scenarios in the same year. For example, in Figure 4, in 2015, agents under trading scenario Neutral (N) have a high year while agents under Profit Driven (PD) have a low performance year (low year). Comparing the scenarios yearly performance from 2005 to 2020, there is no clear cut winner or loser: agents in three different trading scenarios have an equal share of number of high and low years, so no conclusion can be made with regards to the relationship between trading trait and total gain.

Next, we examine the relationship between trait and sale gain, and the relationship between trait and buy gain. Figure 5 shows the yearly average sale and buy gain. The numbers of high and low years for sale and buy were counted. For the high year in sale and buy gain, the count gives us no insight as their probabilities are the same across the board. The result of low year count was represented as pie charts in Figure 5: GD and PD trading scenario have the
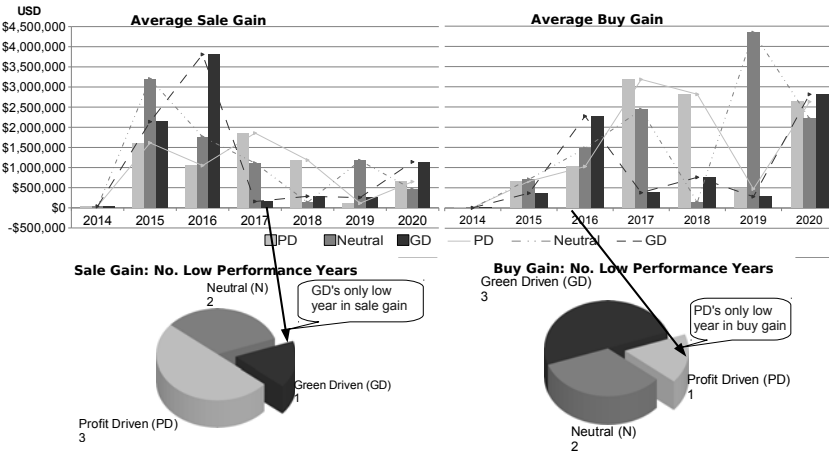
Fig. 5. Yearly Average Sale and Buy Gain in different scenarios: PD vs. Neutral vs. GD

lowest probability (1 out of 6) of encountering a low performance year in sale and buy respectively. With these findings, we can make the following remarks about the relationship between trading trait and sale/buy gain.

From a seller's perspective, adopting environmental trait would likely to yield a sale profit better than average on a yearly basis. With the environmental trait, the probability of having a low performance year is 0.17 (1 out of 6): environmental agents would make high to moderate profit most of the time. On the other hand, if an agent is more likely to be a seller rather than a buyer, profiteering is the worst trait to adopt. Because with profiteering, there is a 50% (3 out of 6) chance of having a low performance in terms of sale gain.

From a buyer's perspective, profiteering is an optimal trait for maximizing buy gain on a yearly basis. In other words, buyer agents should adopt profiteering trait and avoid environmental trait.

In Figure 6, for each experiment, the yearly total gains from 2015 to 2020 are added up and compared to each other. As clearly shown in Figure 6, Neutral trading trait makes the most overall total gain, and fares marginal better for overall sale and buy gain.
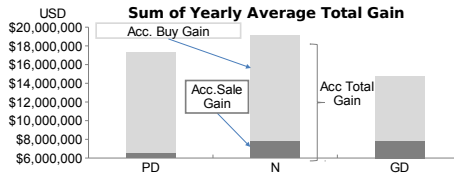
Fig. 6. Sum of Yearly Average Total Gain in different scenarios: PD vs. Neutral vs. GD

5 Conclusions and Future Work

In this paper, we presented a carbon trading model by combining elements of modeling human traits and game theory. The contributions of our carbon trading model to the state of research in this field are twofold. First, we incorporated complex human behaviour in agent. Second, we demonstrated the merits of generalizing strategies by motivations (Profit Driven vs. Green Driven) and Game Players by trading roles (seller vs. buyer). In doing so, we can run simulations under different scenarios. We found that Profit Driven players do not always win. This may be because focusing on only making gains at the expense of others results in no deals: a Neutral trader sometimes earn less but allow deals to pass. Green Driven trading trait perform consistently better than others in terms of yearly average sale gain. On the other hand, Profit Driven trait is best for yearly average buy gain performance. But in terms of total gain performance, Neutral agents - who can swing equally back and forth between Green Driven and Profit Driven trading strategy - earn the most total gain during the last six years of heavy trading (from 2015 to 2020). Based on these results, our recommendation

is Neutral trait which would give agents the highest total gain performance and the second best yearly sale and buy gain performance.

Our results showed that modeling of human trading behaviour in agents creates more market dynamics. In future, we intend to extend the model further by incorporating other human behavioural traits and different socio-economics parameters. The modeling of MAC would also be made for realistic by using polynomial curve fittings technique (which is more accurate than linear fitting).

References

1. Hizen, Y., Kusakawa, T., Niizawa, H., Saijo, T.: Two patterns of price dynamics were observed in greenhouse gases emissions trading experiments: An application of point equilibrium. Tech. Rep. 0557, ISER, Osaka University (2001)
2. McKinsey & Co: Climate change: Everyone's business (2007)
3. McKinsey & Co: Reducing US greenhouse gas emissions: How much at what cost? (2007)
4. McKinsey & Co: An Australia cost curve for greenhouse gas reduction (2008)
5. McKinsey & Co: Greenhouse gas abatement opportunities in Sweden (2008)
6. McKinsey & Co: Pathways to an energy and carbon efficient Russia (2009)
7. McKinsey & Co: Swiss greenhouse gas abatement cost curve (2009)
8. Microsoft: .NET random class. Microsoft Developer Network (MSDN) (2010)
9. Mizuta, H., Kato, S., Tai., H.: Agent-based simulation of CO_2 emissions trading and large-scale traffic flow. In: SICE Annual Conference (2008)
10. Mizuta, H., Yamagata, Y.: Agent-based simulation and greenhouse gas emission trading. In: Proc of the 2001 Winter Simulation Conf. (2001)
11. Mizuta, H., Yamagata, Y.: Transaction cycle of agents and web-based gaming simulation for international emission trading. In: Proc of the 2002 Winter Simulation Conf. (2002), http://www.informs-sim.org/wsc02papers/103.pdf
12. Mizuta, H., Yamagata, Y.: Gaming simulation of the international CO_2 emission trading under the Kyoto Protocol. Agent-Based Social Systems 1, 72–81 (2005)
13. N.Z. Ministry of the Environment: New Zealand's 2020 emissions target (2009)
14. Reeder, J., Sukthankar, G., Georgiopoulos, M., Anagnostopoulos, G.: Intelligent trading agents for massively multi-player game economies. In: Proc. of the Fourth Artificial Intelligence and Interactive Digital Entertainment Conf, AIIDE (2008)
15. Stańczak, J., Bartoszczuk, P.: CO_2 emission trading model with trading prices. Climatic Change 103, 291–301 (2010)
16. Watkins, C.: Carbon emissions markets: A simulation approach. In: The Australian National University International Congress on Modelling & Simulation Proc. (2001)

Fast and Incremental Neural Associative Memory Based Approach for Adaptive Open-Loop Structural Control in High-Rise Buildings

Aram Kawewong[1,2], Yuji Koike[3], Osamu Hasegawa[1], and Fumio Sato[4]

[1] Imaging Science and Engineering Laboratory, Tokyo Institute of Technology,
4259-R2-52 Nagatsuta, Midori-ku, Yokohama, 226-8503, Japan
[2] Department of Computer Engineering, Faculty of Engineering, Chiangmai University,
Chiang Mai, 50200, Thailand
[3] IHI Corporation, Research Laboratory,
1, Shin-nakahara-cho, Isogo-ku, Yokohama, 235-8501, Japan
[4] IHI Corporation, Infrastructure Operation,
Toyosu IHI Building, 1-1, Toyosu 3-chome, Koto-ku, Tokyo 135-8710, Japan
{kawewong.a.aa,hasegawa.o.aa}@m.titech.ac.jp,
{yuuji_koike,fumio_satou}@ihi.co.jp

Abstract. A novel neural associative memory-based structural control method, coined as AMOLCO, is proposed in this study. AMOLCO is an open-loop control system that autonomously and incrementally learns to suppress the structural vibration caused by dynamic loads such as wind excitations and earthquakes to stabilize high-rise buildings. First, AMOLCO incrementally learns the associative pair of input excitation from either winds or earthquakes and the corresponding output control response generated by standard optimal control only under a single simple condition (i.e., low wind conditions). After learning for a short period of time, i.e., 15 min, AMOLCO becomes capable of efficiently suppressing more intense structural vibrations such as those caused by very strong winds or even earthquakes. In this study, evaluation of the AMOLCO method is performed by using the physical simulation data. The results show that the control signal generated by AMOLCO is similar to that generated by the state-of-the-art control system used in a building. In addition, the resulting control signal is tested on a realistic simulation to affirm that the signal can control the structures. These results show that for the first time, AMOLCO offers another approach of structural control, which is inexpensive and stable similar to a standard open-loop system and also adaptive against disturbances and dynamic changes similar to a closed-loop system.

Keywords: Self-organizing neural network, Active-tuned mass damper, structural control, open-loop control.

1   Introduction

For nearly half a century, structural vibration control has been a topic of intensive research in both civil engineering and the architecture industry. The
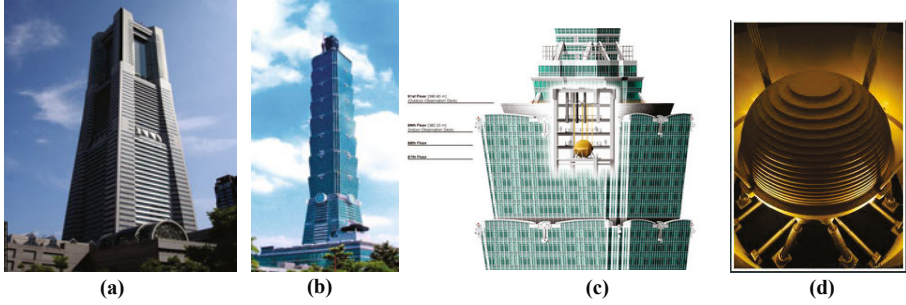
Fig. 1. Buildings with the mass damper installed for structural control. (a) The landmark tower of Japan. (b) The Taipei 101 in Taiwan. (c) The illustration of how the damper has been installed into the Taipei 101 building. (d) The damper of Taipei 101 building.

control of this parameter is indispensable for public safety, which is always of utmost importance. One of the most well-known structural control strategies is the use of a mass damper (Fig. 1 (d)) that is installed on the top floor of a building (Fig. 1 (c)) to stabilize the structure against dynamic loads such as wind and earthquake. The device consists of a mass, spring, and damper and is attached to a structure to reduce its dynamic response [1]. Some examples buildings with mass dampers installed are Taipei 101 (Fig. 1(b)) in Taiwan, Yokohama Landmark Tower (Fig. 1 (a)), and Trump world tower in the USA.

From the structural control perspective, the control method of the mass damper is categorized into four main paradigms: *active, passive, hybrid* and *semi-active* [2]. In passive control, the damper remains functional without an external power source, posing no significant risk of an unstable situation. However, the devices are incapable of adjusting to structural changes and varying loading conditions [2]. In contrast, active control is expected to provide more effective protection against winds and earthquakes by utilizing a variety of computer-controlled mechanisms with smaller mass weights. Thus, passive control has advantages of high stability and reliability with low cost and low power requirement, while active control offers adaptability and versatility [3], [4]. Hybrid control is a combination of passive and active controls [5]. Finally, semi-active control is actually a passive control that functions with low power requirement while achieving the active control performance [6].

Although research in active structural control is popular [1], [3], [6], there are still a number of challenges including (1) reduced capital cost and maintenance and (2) increased system reliability and robustness. This paper particularly addresses these two challenges. In this paper, neural associative memory is applied to these challenges. We assumed that the damper's movement and the building's input excitation had a pattern. Thus, a set of associative neurons are created by first observing the real building controller. Once a sufficient number of associations were learned, the proposed neural associative memory-based open-loop control system (AMOLCO) performed structural control by deriving the input excitation and outputting the appropriate response from its associative memory. The associative memory storage can be updated incrementally. The evaluation was

using the simulation data that features many challenging vibration conditions such as earthquakes and strong winds. We found that the control signal generated by AMOLCO was similar to that generated by the state-of-the-art optimal control method of such as Linear Quadratic Regulator (LQR) [7] and the H_∞ technique [3]. Another realistic simulation was also conducted to show that the output control of AMOLCO can suppress the vibration of the building.
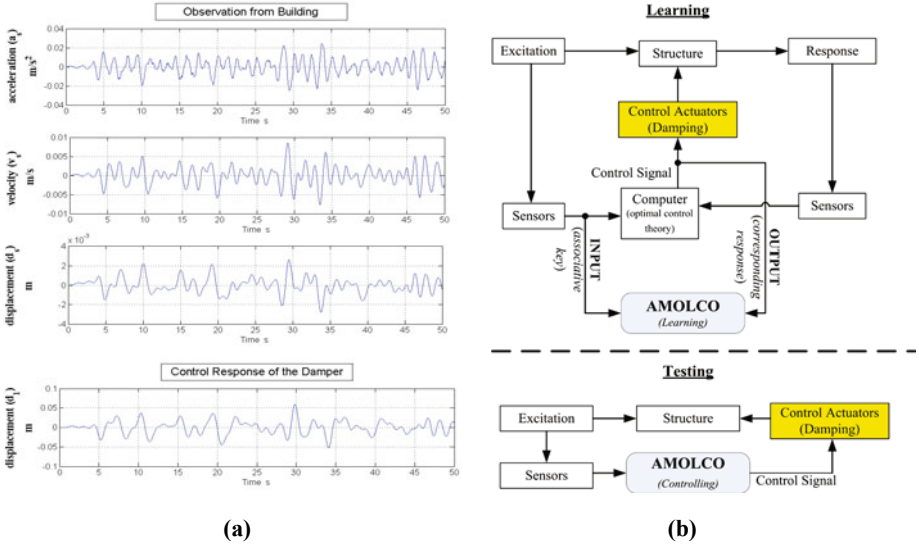
Fig. 2. (a) Sample of data observed from structural control during 50 s period of time. (b) The overview of the proposed AMOLCO system for structural control.

2   Problem Definition and Related Works

Modern control systems can be classified as either open-loop or closed-loop. In an open-loop control system, the control action is independent of the output (no feedback). In a closed-loop control system, the control action depends on the feedback, resulting in the risk of oscillations in the output. As a result, for open-loop control systems, oscillations are not possible, and thus, the method lacks the adaptability to perform new control actions corresponding to unfamiliar conditions.

Many previously proposed active control approaches are based on different control strategies such as Fuzzy Control [8], Optimal Control [3], [7], or Neural Networks [9]. Nevertheless, closed-loop controllers still have a risk of instability since the output control signal depends on the feedback from the building. Therefore, this paper proposes a new control system that is stable similar to the open-loop controller and also adaptive to dynamic changes similar to the closed-loop approach. Fig. 2 (b) portrays an overview of the proposed system, AMOLCO. First, the system self-organizes and incrementally learns from the closed-loop control system (right) for a short period of time. Then, it controls the building in an open-loop fashion.

Before discussing AMOLCO in detail, some essential input variables for structural control must be clarified. For input excitations, *three* variables were observed from the sensors mounted on a building. For output control, *one* more variable was used. Note that the structural control addressed herein is allowed of single degree of freedom. All variables are described as follows:

a_s : acceleration of the structure
v_s : velocity of the structure
d_s : displacement of the structure
d_m : command displacement of the mass damper

AMOLCO must capture the associations between a_s, v_s, d_s and d_m and represent them in an appropriate compact format. Fig. 2 (a) portrays the sample data of four variables during 50 s observation time. Three variables describe the status of the building and one describes the damper movement.

Reviews of the literature about the application of associative memory indicate that there is a clear difference between the proposed AMOLCO system and the fuzzy associative memory (FAM) approach [10]. Most of the FAM approaches [10], [11] require a construction of interpretable and meaningful fuzzy sets, whereas AMOLCO does not have such requirement. FAM aims to model the association among fuzzy rules, while AMOLCO looks for associations among real values. That is, unlike FAM, in AMOLCO, the associations are represented in the space of $\mathbb{R}^n$, where n is the dimension of the feature vector (section 3).

The objectives of AMOLCO and FAM approaches are different. The FAM-based method is essentially an improvement of the fuzzy-logic-based method. Both the methods based on fuzzy logic neural networks-based methods are intentionally proposed to replace the current optimal control theory [3], [7]; it does not learn from the optimal control theory. However, the main purpose of these methods is to reduce human efforts required for designing analytical control theories such as H_∞ [3] and LQR [7]. Unfortunately, the performance of the intelligent closed-loop control strategy is still not sufficient for real use. Moreover, these approaches are still considered a kind of closed-loop system. However, the proposed AMOLCO aims to learn the best control strategy that can be applied in real buildings, so as to generate very similar control actions corresponding to the input excitations. It should be emphasized that AMOLCO needs to learn from only *one* simple condition (small winds that occur once in 0.1 years) for only a short period of time (900 s in this paper), and is capable of controlling the building under more intense conditions such as strong winds that occur once in 1000 years and earthquakes.

3   Proposed Approach

The proposed AMOLCO has two phases: learning and controlling (testing). Being online and incremental, AMOLCO can switch forth and back between these two phases at anytime. Fig. 2 (b) shows an illustration of the working of AMOLCO.

3.1   Feature Representation

We represent associations in the form of "if INPUT then OUTPUT," where INPUT is an *associative key* (input excitation) and OUTPUT is the *corresponding output*

(control response). To generate the appropriate control action, AMOLCO needs to process the input data *periodically*. In particular, given ω as the length of the interval of the time series input data for consideration, $\mathbf{a}_s$, $\mathbf{v}_s$, $\mathbf{d}_s$ as variables observed up to time t, i.e., $\mathbf{d}_s = ((d_s)_1, \dots, (d_s)_t)$, the feature vector of associative key at time t is

$$\mathbf{x}_t = \left(x_t^{(1)}, x_t^{(2)}, \dots, x_t^{(\omega)} \right) \tag{1}$$

The variable $x_t^{(i)}$ might be displacement $(d_s)_{k+j}$, acceleration $(a_s)_{k+j}$ or velocity $(v_s)_{k+j}$ of the structure, where $1 \leq j \leq \omega$, $1 \leq k \leq t - \omega + 1$, and i indicates the element index of the vector. The parameter ω affects the performance of the AMOLCO. If ω was too large, then the association would not be sufficiently discriminative to produce the accurate control response. In contrast, if ω was too small, the information in the associative key would be insufficient to correctly represent the input excitation. The input associative key $\mathbf{x}_t$ comes with a corresponding control response $\mathbf{y}_t$,

$$\mathbf{y}_t = \left(y_t^{(1)}, y_t^{(2)}, \dots, y_t^{(\omega)} \right) \tag{2}$$

where $y_t^{(j)}$ is $(d_m)_{k+j}$ from the variable $\mathbf{d}_m$ observed up to time t. Then, an *associative pattern* $\mathbf{z}_i$ is derived by

$$\mathbf{z}_t = \begin{bmatrix} \mathbf{x}_t \\ \mathbf{y}_t \end{bmatrix} \tag{3}$$

where $\mathbf{x}_t, \mathbf{y}_t \in \mathbb{R}^\omega$.

One problem of concern is that $\mathbf{x}$ and $\mathbf{y}$ might be of different units (i.e., acceleration $\leftrightarrow$ displacement). This may lead to errors in clustering because the distance is measured, Euclidean distance, where all dimensions of the vectors bear the same importance and priority. For example, associative pairs between a_s and d_m have different unit scales. The scale size of d_m is larger, so the data would be grouped into clusters primarily on the basis of d_m. Therefore, we create the associative pair of only the *same* units, i.e., velocity with velocity and displacement with displacement. Namely, out of the three possible types of associations, only $d_s \leftrightarrow d_m$ are used. Note that we use the notation $F \leftrightarrow R$ to represent the association between the associative key F and the corresponding output R.

3.2 Online Incremental Neural Associative Memory [12]

Self-organizing and incremental neural associative memory, coined as SOIAM [12], is a recently developed incremental online algorithm which that can learn from associative input data in an incremental manner. Its performance has been reported to be better than that of bidirectional associative memory (BAM) [13], Hopfield-based methods [14] and the Kohonen feature map [15]. Starting from the empty set, SOIAM considers the first two input data as the starting two nodes, and then, for every input associative pattern $\mathbf{z} \in \mathbb{R}^{\omega+\omega}$, where $\mathbf{x} \in \mathbb{R}^\omega$ is an associative key and $\mathbf{y} \in \mathbb{R}^\omega$ is a corresponding output, the algorithm finds the first-nearest node s_1 and the second-nearest node s_2 by the equations:

$$s_1 = \text{argmin}_{c \in \mathbf{A}} \| \mathbf{z}_t - \mathbf{W}_c \| \tag{4}$$

$$s_2 = \text{argmin}_{c \in \mathbf{A} - \{s_1\}} \| \mathbf{z}_t - \mathbf{W}_c \|, \tag{5}$$

where **A** is the set of all nodes in the SOIAM and $\mathbf{W}_c$ is the ω-dimensional weight vector of node c. If the distances between the new input associative pattern and the first and second winners are less than the threshold, the pattern is assigned as the first winner node. Otherwise, SOIAM will determine that the input associative pattern is very different from the current nodes and that a new node should be created.

Consider the case in which a new associative pattern is assigned to the nearest node s_1 in the SOIAM; the weight vector $\mathbf{W}_{s_1}$ will be updated by the value of the new input pattern and an edge between the first and the second winners will be created (if it does not exist already). The clustering behavior of SOIAM [12] and SOINN [16] are similar, and both are significantly different from other clustering methods such as k-means. The new input associative pattern data is not directly added to form the cluster. Instead, the cluster is formed by connecting existing SOIAM nodes. This saves a significant amount of running memory in the long term. The SOIAM also allows the nodes to be autonomous agents. At any time, each node has its own age and accumulated errors (represented by the sum of the distance of the input pattern for every time it is selected as the first winner). With these properties, each node can perform two activities: it can become dead or it can divide itself. If the node exists for a long time without winning for any new input pattern (a noisy or useless node), all connected edges will gradually die. Alternatively, if the accumulated noise is very large, the node will divide itself into two.

3.2.1   Learning the Associations (Training)

To obtain informative data, the sliding window technique with the shifting steps s has been employed. That is, the number of input data we can obtain at time t is $t - \omega + 1$ ($s = 1$). We have tried different values of s, but they do not affect the accuracy of AMOLCO significantly. The parameter s significantly affects only the computation time since it indicates the amount of data available for learning. We use $s = 5$ for all experiments in this paper. This value is small enough to enable AMOLCO to respond in less than 0.01 s (data sampling time of a real building).

In this paper, we use the training data obtained under very low wind conditions that occur frequently (every 0.1 years) for 900 s with a sampling rate = 0.01 s. This results in data of a total of $\left(((90000) - 200)/5\right) + 1 = 17961$ associative pairs that have been used as the input for SOIAM learning. The parameter ω is set by SOIAM to allow the pattern to capture at least two *local optimum points* in the input wave ($\omega = 200$ in this paper). A value of more than 200 does not significantly increase the algorithm accuracy but slows down AMOLCO so that the system cannot respond in the desired response time.

3.2.2   Controlling the Structure (Testing)

At any time, AMOLCO can perform structural control by using the associative memory learnt so far. However, all previous learnings have been done in only one simple condition. Therefore, it is very unlikely that the system would be able to control the structure in different conditions, i.e., during an earthquake. As a solution, we use a simple but efficient technique to amplify the magnitude of the nodes. This technique is based on our observations that the pattern of the input associative key and its corresponding output is stable across different environments. The only

difference is the magnitude of the input and output. Therefore, duplicating SOIAM nodes into multi-scale clusters could enable AMOLCO to control the building in variable conditions. In this study, we multiply the clusters in 5 scales. That is, given $n(\mathbf{A})$ as the number of all nodes in SOIAM at the current time t, the multiplication process results in a total of $n(\mathbf{A}) \leftarrow n(\mathbf{A}) + 5 \times n(\mathbf{A})$ nodes in SOIAM. Note that these multiplied nodes are used only during the structural control phase. If the system switches back to the learning mode, all the duplicated nodes are discarded.

AMOLCO obtains the time series input associative key, $\mathbf{x}_t$, and then retrieves the appropriate response, $\mathbf{y}_t$, from a set of associative nodes from the three neural associative memory systems corresponding to the associations $\mathbf{A} = \{s_1, \dots, s_{n(\mathbf{A})}\}$, where s_i is the ith node of the node set $\mathbf{A}$ of the neural associative memory of the association between d_s and d_m at current time t. In particular, the desired corresponding displacement control for damper (d_m) is derived by using a query with the input associative key

$$\mathbf{x}_t = ((d_s)_{t-\omega+1}, \dots, (d_s)_t) \tag{6}$$

to retrieve the set of nodes in the neural associative memory $\mathbf{y}_t = \mathbf{r}_i$, where

$$i = \text{argmin}_{1 \le i \le n(\mathbf{A})} \|\mathbf{x}_t - \mathbf{f}_i\| \tag{7}$$

$$\mathbf{W}_i = \begin{bmatrix} \mathbf{f}_i \\ \mathbf{r}_i \end{bmatrix} \tag{8}$$

4   Experiments and Results

To test the performance of the proposed method, AMOLCO, we use physical simulation dataset to feature many challenging vibration conditions. Since AMOLCO is the first control system approach used for controlling structure, which works in an open-loop fashion by using the associative memory, there is no established baseline for comparison. Thus, we also utilized AMOLCO based on *k*-means to compare the results with the SOIAM-based AMOLCO approach.

The data was obtained from realistic physical simulation of a 20 m tall building. It models five different wind conditions and two different earthquakes. Each dataset was studied for only 1000 s. The data sampling rate is 0.01 s, so a total of 100,000 data points per condition were obtained. The detail of the datasets is shown in Table 1. The return period year of the wind (i.e., 0.1, 1, 10 years) is the period in which the wind occurs *once*. For instance, the wind condition of 1 year indicates that the wind occurs once in 1 year. Ultimately, wind speed experienced by a building depends on its location. In this paper, the wind speed is assumed to range from about 21 to 35 m/s.

Table 1. Description of datasets

Conditions	data length (s)	Sampling rate (s)	Train data
wind 0.1 year	1000	0.01	
wind 1 year	1000	0.01	
wind 10 year	1000	0.01	wind 0.1
wind 50 year	1000	0.01	(1~900s)
wind 1000 year	1000	0.01	
Hachinohe EW earthquake	100	0.01	
Elcentro NS earthquake	100	0.01	

Out of the seven trial conditions, only the condition with 0.1 year wind condition is used to train AMOLCO. The training was done in an online incremental manner; the signal was *incrementally* input to AMOLCO every 0.01 s. Then, AMOLCO began to control the building structure under various excitation conditions.
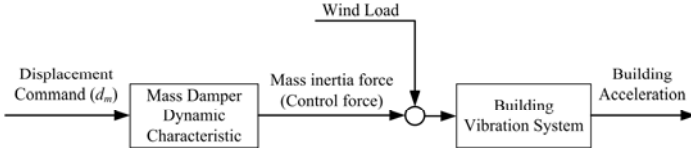
Fig. 3. Signal flow for calculating the building acceleration using the mass damper displacement command generated by AMOLCO

For evaluation, we compared the performance of k-means-based and SOIAM-based AMOLCO. The performance of each method is evaluated by considering the root mean square (RMS) error compared to the ground truth (original optimal LQ control [7]). The comparisons are done for both command displacement and building acceleration. The comparison of the d_m signal with that of the ground truth shows the similarity of displacement command. In Fig. 4 (right column), it is clear that the damper displacement command of AMOLCO is very similar to that of the ground truth. In addition, we create another simulation to test if AMOLCO's control signal can adequately control the building. The model of this simulation is shown in Fig. 3. The building acceleration response is analyzed by using its signal and comparing it to the acceleration response of the ground truth. This figure shows the case for wind load excitation. The mass inertia force is obtained by inputting the displacement command into the mass damper dynamic characteristic. The inertia force acts to suppress the wind load. In case of an earthquake, the building acceleration is calculated by considering the wind load as the inertia force induced by the ground motion.

The results suggest that both k-means-based and SOIAM-based AMOLCO can efficiently control the building even though the test conditions are significantly different from the training conditions; the system learns to control the building in small winds and can achieve suppression of vibrations induced from earthquakes. The RMS error between 8 and 15 % is considered adequate for real construction. Fig. 4 shows a graph of a typical output signal sample of AMOLCO and the ground truths. The signal is similar for both command and building acceleration.

Both k-means-based and SOIAM-based AMOLCO show similar performance. Nevertheless, k-means has many drawbacks: (i) the computation of k-means data clustering is expensive. (ii) the number of k must be manually input by a user. (iii) k-means cannot learn new data incrementally. We conduct another experiment to show that increasing the number of data learning period results in an increasing accuracy of AMOLCO. This shows that for the future work of fully autonomous control system that learn to control all by itself from the beginning, where training data would be input incrementally, the SOIAM-based AMOLCO is more suitable since its accuracy will increase with an increasing amount of training data. This is shown by the graph in Fig. 5 (b). The graph shows the accuracy of SOIAM-based AMOLCO after exposure to different number of training datasets. It is clear that the accuracy of

Table 2. Root mean square error of building acceleration of AMOLCO

Method	RMS Error of building acceleration (%)							Avg. Error	Learning time (s)
	Wind excitation (return period years)					Earthquakes*			
	0.1	1	10	50	1000	elcentro	hachiew		
SOIAM	13.50	14.20	14.30	15.20	9.29	3.85	4.88	**10.75**	**115.11**
k-means	17.08	20.79	11.79	11.78	5.37	7.58	4.27	11.24	463.97

* elcentro = Elcentro NS, hachiew = Hachinohe EW
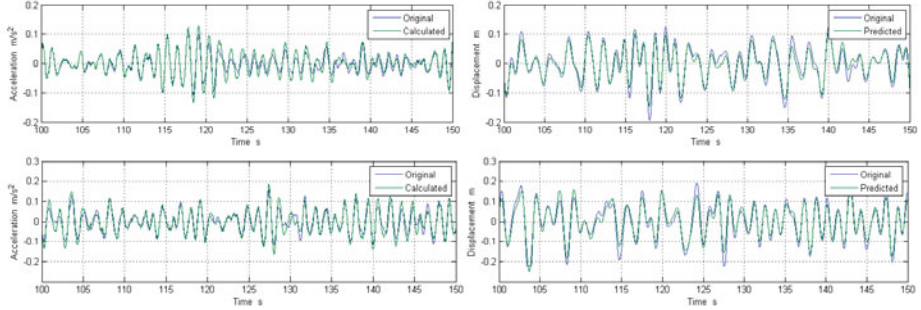
Fig. 4. Results of the experiment. (**Left column**) Graphs show the building acceleration under wind excitation at 50 and 1000 return period years, respectively, controlled by the SOIAM-based AMOLCO and LQ optimal control [7]. (**Right column**) Graphs show the signal of damper displacement command generated by SOIAM-based AMOLCO and the ground truth (LQ control [7]).
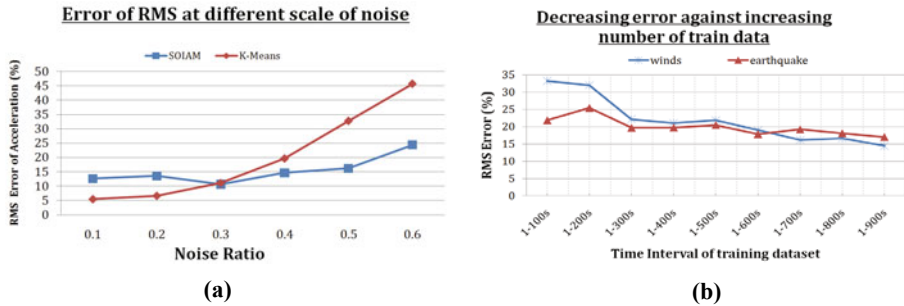
Fig. 5. (a) RMS value errors of the SOIAM-based and k-means-based AMOLCO at different noise scales. The SOIAM-based method is much more robust against noise than the k-means method. (b) The decreasing value of mean error of the displacement command of SOIAM-based AMOLCO.

AMOLCO increases with the increasing number of training periods. (iv) SOIAM is very robust against noise. We conducted an experiment by randomly adding noise to the training dataset (wind 0.1 [1~900]). In particular, we intentionally paired the associative keys with the wrong corresponding output sampled from the dataset itself. This simulates the situation where the system learns to control the system through trial and error (future work). The wrong associative patterns added to the training set were 10%, 20%, 30%, 40%, 50% and 60% in each case. Each case was repeated 10 times. The result of this noise simulation is shown in Fig. 5 (a).

5   Conclusion

By the results, AMOLCO successfully handle the challenges stated previously. It is an inexpensive structural control that poses no risk of output oscillation while being adaptive to deal with unfamiliar intense vibration conditions. The simulation in Fig. 3 allowed us to the graph plotted in Fig. 4 (left-column) to confirm that the control output from AMOLCO can actually stabilize the building and make the acceleration as close to zero as possible. It is also noteworthy that the final goal for the AMOLCO project is to be a fully self-adaptive system that learns to control the building structures all by itself right from the beginning. The version of AMOLCO in this paper is an important step in this process, which proves that SOIAM-based AMOLCO can accurately learn in an incremental manner from very noisy associations, a characteristic that is highly suitable to the reinforcement learning scheme.

References

1. Kobori, T., et al.: Seismic response controlled structure with active mass driver system. Part 1: Design. Earthquake Eng. Struct. Dyn. 22, 133–139 (1991)
2. Spencer, B.F., Nagarajaiah, S.: State of the art of structural control. Jour. Struct. Eng. 129, 845–856 (2003)
3. Alkhatib, R., Golnaraghi, M.F.: Active structural vibration control: A review. The Shock and Vibration Gigest 35(5), 367–383 (2003)
4. Fujinami, T., et al.: A hybrid mass damper system controlled by H^∞ control theory for reducing bending-torsion vibration of an actual building. Earthquake Eng. Struct. Dyn. 30, 1639–1653 (2001)
5. Koike, Y., et al.: Application of V-shaped hybrid mass damper to high-rise buildings and verification of damper performance. In: Struct. Eng. World Conf., SEWC, T198-4 (1998)
6. Ikeda, Y.: Active and semi-active control of buildings in Japan. Jour. Japan Asso. Earthquake Eng. 4(3), 278–282 (2004)
7. Ribakov, Y., Reinhorn, A.M.: Design of amplified structural camping using optimal considerations. Jour. Struct. Eng. 129(10), 1422–1427 (2003)
8. Guclu, R., Yazici, H.: Vibration control of a structure with ATMD against earthquake using fuzzy logic controllers. Jour. Sound and Vibration 318(1-2), 36–49 (2008)
9. Yang, S.M., et al.: Structural vibration suppression by a neural-network controller with a mass-damper actuator. Jour. Vibration and Control 12(5), 495–508 (2006)
10. Chung, F.-L., Lee, T.: On fuzzy associative memory with multiple-rule storage capacity. IEEE Trans. Fuzzy Syst. 4(4), 375–384 (1996)
11. Sussner, P., Velle, E.: Implicative Fuzzy Associative Memories. IEEE Trans. Fuzzy Syst. 14(6), 793–807 (2006)
12. Sudo, A., et al.: Associative memory for online learning in noisy environments using self-organizing incremental neural network. IEEE Trans. Neural Netw. 20(6), 964–972 (2009)
13. Kosko, B.: Bidirectional associative memories. IEEE Trans. Syst. Man Cybern. SMC-18(1), 49–60 (1988)
14. Hopfield, J.J.: Neural networks and physical systems with emergent collective computational abilities. Proc. Nat. Acad. Sci. USA 79(8), 2554–2558 (1982)
15. Yamada, T., et al.: Sequential learning for associative memory using Kohenen feature map. In: Proc. Int. Joint Conf. Neural Netw. (1999)
16. Shen, F., Hasegawa, O.: An incremental network for on-line unsupervised classification and topology learning. Neural Netw. 19(1), 90–106 (2005)

Emergence of Leadership in Evolving Robot Colony

Seung-Hyun Lee, Si-Hyuk Yi, and Sung-Bae Cho

Dept. Computer Science, Yonsei University,
50 Yonsei-ro, Seodaemun-gu, Seoul 120-749, Korea
{e2sh83,theshy}@sclab.yonsei.ac.kr, sbcho@cs.yonsei.ac.kr

Abstract. Growing interest in robot colony has led to initial experimental applications in biology, sociology, and synecology. Especially, it is noticeable that some researchers have tried to study on robot colony using evolutionary computational. In this paper, we present an evolutionary robot colony model and analyze their behavior for leadership characteristics in group of robots. Each robot has its own social position: leader, follower, and stranger. Leaders have responsibility of the existence of its group while followers choose their behavior going after their leaders'. Strangers behave independently without a leader or a follower. Transition between social positions is controlled by simple rules and probability, and behaviors change adaptively to the environment using evolutionary computation. Simulation has been conducted with 2-D based robot simulator Enki of EPuck mobile robots. Through experiments, we have found that the more centralized structure emerges in the evolutionary robot colony with a few leaders and safety behavior policy when facing with a difficult condition.

Keywords: Evolutionary computation, mobile robot colony, leadership behavior.

1 Introduction

Recently, multiple mobile robots based analysis method is actively applied for different research areas. In the initial stage, more efforts were put on making robots intelligent so that they can choose their own behavior adaptively to the environment like robot soccer field [1]. This has been connected to a research area that tries to learn from multiple robots' behavior rather than the reproduction itself such as communication and information suppression between robots [2], [3]. The aim of this sort of research is to explain phenomena and to get insight from their behavioral development. In this respect, studying emergence of leadership behavior and development of social structure in robot colony is also one of interesting and untreated topics.

Krause defined leadership as the emergence of new behavioral pattern by some entities that are promptly accepted by other group members [4]. Analogous to Krause, Robbins describes leadership as the exerting one's influence to other entities for achieving a specific goal [5]. So we can redefine that leadership is a power of influence that exert voluntary acceptance to other entities for a definite aim with a

new behavior strategy. This leadership was steadily observed from not only human beings but also animal colony such as primates, canids, birds and fishes [6]. Some researchers extended to build a computational model that captures a leadership behavior from colony's moving.

In this paper, we investigate the emergence of leadership behaviors and developmental aspects from mobile robot colony with evolutionary behavior. Robot can have three different social status, leader, follower, and stranger, and adaptively use this social relation to survive in the competitive world. The robot's behavior is manifested by evolutionary neural network which is the source of the development of new behavior. We design two experimental groups based on the level of risk: high-risk and low-risk assignments. From the independent simulation results, we analyze the developmental aspects of social structure and leadership behavior, and also investigate which environmental characteristics are favorable for the emergence of leadership in mobile robot colony.

2   Related Works

Research on the evolutionary robot colony can be divided into two areas based on the main focus: one is research on the activity and the other is communication between robots. For an example of the research on communication, Floreano et al. analyzed how robot colony develops the way of communication when they found food or poison with embedded LED sensor [2]. They utilized neural network for the control of robot's movement, and experiment with different selection method and grouping mechanism. As an extension to Floreano's study, Mitri et al. found that robots deceive other entities when their interests conflict with each other [3]. This was shown by the emission of light nearby poison.

On the other hand, diverse characteristics that lead leadership also were analyzed in animals, human beings and even digital organism. Cozin et al. indicated that the difference in information each animal has is the primary motive for the pattern of mass movement [7]. Similarly, situation of neighbor entity was known to the key factor which decides an entity's movement in its group [8]. As well as the findings of animal behavior, Dyer et al. revealed that the size of group, spatial structure of the group, and the a few entities who have information decide the emergence of leadership behavior in human groups. Especially, they suggested that groups can achieve the goal when a leader is located in the center or the each corner. Last, selection method can make a difference in leadership behavior from the experiment of digital structure [9]. Based on these previous researches that dealt with leadership and mobile robots, we focus on the developmental aspect of leadership behavior given two different settings.

3   Modeling Evolutionary Robot Colony

3.1   Overview

The proposed simulation model consists of three parts as shown in Fig. 1: socializing module, artificial working space, and evolution engine. Socializing module creates and eliminates small robot group and changes each robot's social status based on the simple

predefined rules in a stochastic way. These social relations between each mobile robot are inputted in the artificial simulation space. In the artificial working space, each colony and individual robot competes with each other to achieve a goal which is realized as hunting a prey robot. Based on the simulation results, a child robot with new strategy created from excellent parents robots is generated and replaced with the one eliminated from the robot colony. This evolutionary computation is controlled by evolution engine with selection, crossover, and mutation operation.
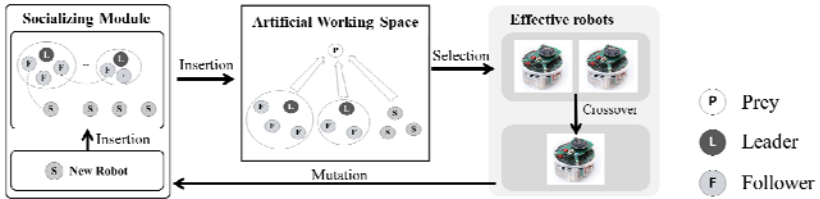
Fig. 1. Evolutionary robot colony framework

3.2   Social Position

Mobile robots in the proposed model can take a social position among leader, follower, and stranger. Leaders are responsible for the survival of its own colony. Leaders decide their own behavioral strategy which affects the followers' behaviors. No robots are allowed to become a leader without at least follower. Followers choose their behavior based on the changes in environment and their leaders. Strangers do not belong to any colony. Strangers act by themselves not considering any other robots similar to leaders. However, strangers do not take responsibility nor gain any help from other entities. Robots can choose collective or independent activity considering their situation. Initial social position is set to a stranger for all robots.

Group G consists of a tuple, a leader L and a set of follower F, $G=<L, F>$. We assume only a leader can take a group. So given G has at most one leader, $n(L) = 1$, and at least one follower, $n(F)>=1$. Group is broken up when a leader is eliminated or no follower exists. A follower can follow only one leader at a time, but the change of its leader is allowed.
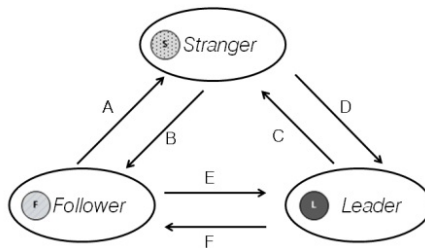
Fig. 2. Six possible changes in social position. (A) Elimination of the leader robot. (B) Incapable of self-sustenance. (C) Elimination of all followers. No followers. (D) Emergence of a follower. (E) Get higher efficiency than the leader. (F) Emergence of a more efficient follower.

Fig. 2 depicts all possible changes of robots' social position. This transition divided into voluntary transition and compulsory transition. Voluntary transition happens for taking advantage of survival such as being a follower when no sustainable capability, and being a stranger when a leader is incapable. On the other hand, compulsory transition happens due to the structural changes like group's or robot's extinction.

We need to decide a guideline for some transitions of social position. For example, a robot should be able to decide when it follows more efficient entity. A robot also should be able to choose when it stops following or changes a leader. For this purpose, we design a transition mechanism in a stochastic way. An index that can guarantee appropriate decision is vitality of a robot entity in the artificial world. When vitality Vital is given, we can calculate the transition probability from stranger to follower is as (1) where n_n is the size of colony.

$$P_{follow}(i) = 1 - \frac{Vital_i}{\sum_{r=1}^{n} Vital_r / n_n} \tag{1}$$

Even if a robot decides to be a follower, it does not always guarantee an organization of a group. Becoming a leader can also be affected by the current situation, the amount of difference in vitality between the potential follower j and the potential leader i. The probability of emergence of leader is defined as (2) by considering the relative vitality gap between i and j.

$$P_{leader}(i, j) = \frac{Vital_i}{\sum_{r=1}^{n} Vital_r} \frac{Vital_j - Vital_i}{MAX_{vital} - MIN_{vital}} \tag{2}$$

When following the current leader is not beneficial for its survival, a robot can change a leader or be a stranger. The probability a robot changes its leader is decided based on its vitality and average vitality of members in the other groups as (3) where n_k, n_j is the colony size of k, j, respectively.

$$P_{change}(i, j, k) = \frac{Vital_i}{\sum_{r=1}^{n} Vital_r / n_n} \frac{\sum_{r \in k}^{n(k)} Vital_r / n_k}{\sum_{r \in i}^{n(j)} Vital_r / n_j} \tag{3}$$

If the calculated probability is lower than threshold, a follower changes its social position into stranger.

3.3   Behavior Control and Evolution

Mobile robots' behavior pattern is realized by the control of speed of both left and right wheels. So it is important to define an adjusting mechanism of wheel speed. In this work, we use a neural network model for dynamic and flexible behavior strategy as shown in Fig. 3.

Neural network model consists of 8 input nodes, 2 output nodes and 16 weights which connect inputs to outputs. All robots use distance, relative location to prey robot and velocity of prey robot on x and y axis as inputs. Follower robots additionally get distance, relative location, velocity of a leader robot. These inputs about a leader and weights are disregarded by leader and stranger robots in the calculation of its wheel speed. Each observed input values are normalized based on its
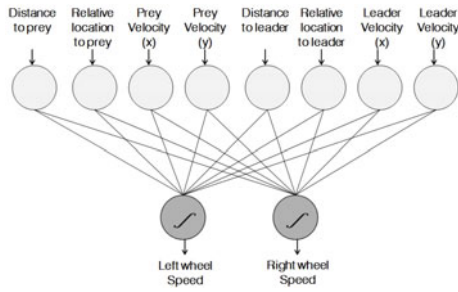
Fig. 3. Structure of neural network for control of robot behavior

maximum and minimum values. In the output nodes, the output is filtered by sigmoid function and actual wheel speed is calculated by multiplying filtered value and maximum wheel speed.

We achieve robots' adaptive behaviors by using the basic evolutionary computation operations such as selection, crossover, and mutation [10]. The search space is the weights value between input and output nodes. We do not invite the structure of neural network, but we try to find appropriate weights in the fixed simple network structure.

First, entities which are efficient within 50% based on fitness value are selected for the generation of new strategy. We used 3-point crossover to create a new strategy from parent strategies which is for a robot that is inserted in the artificial world. The reason for the application of 3-point crossover is to minimize the effect from the difference by social status. Finally, through mutation new strategy is finalized. Mutation is also applied for the strategy of less efficient robot entities in order to lead positive changes.
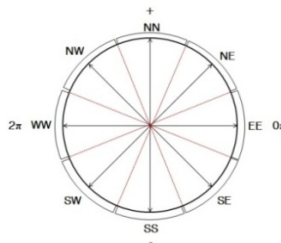
Fig. 4. Eight directions in the artificial world

4   Experimental Design

4.1   Assignments Design

The goal of mobile robot with two wheels is to survive from the artificial 2-D simulator by hunting a prey robot P. As described above, each robot cooperates or competes according to its social position. Robot is randomly placed with the same

distance from *P* located in the center regardless of its group or social position. Robot has a head which becomes a direction of vision field. The direction is divided into 8 subareas represented in real number depicted in Fig.4. NE, NN, and NW have positive value while SW, SS, and SE have negative.

Physical collision in the given time determines success or failure in the designed hunting situation. The robot entity that touches *P* takes possession from the successful hunting. If this robot belongs to a group, prey is shared with the members; a leader takes large proportion. If a stranger catches *P*, it takes all.

Robots sometimes are injured in the hunting process. In case of head-on collision with the prey, robots get damage from injury. When a robot bumps into *P*, the angle of moving directions determines the extent of injury described in (4).

$$Injury(i) = \begin{cases} Injury_{HIGH} & ,If\ Dir(i) + Dir(p) \equiv 0 \\ Injury_{LOW} & ,If\ Dir(i) + Dir(p) \equiv \pm 1 \end{cases} \tag{4}$$

Fitness value Fitness of entity *i* is defined as (5)

$$Fitness(i) = V_{success} + \frac{V_{group}}{Dist(i,p)} - Injury(i) + \frac{Time_{rest}}{Dist(i,p)} \tag{5}$$

where *Dist(i, p)* indicates the distance from *P* to *i* where $V_{success}$ and V_{group} mean the gain when it or its group member successes. Both $V_{success}$ and V_{group} are activated in case of success, and $V_{success}$ is always higher than V_{group}. Fitness value increases robot's vitality while the default cost for living $Energy_{cost}$ decreases it as shown in (6).

$$Vitality(i) += Fitness(i) - Energy_{cost} \tag{6}$$

When vitality has negative value, the robot is eliminated from the artificial space with its social position. Stranger robot with new behavior strategy replaces the elimination.

4.2   Settings

We design two different experimental groups according to the difficulty of the given assignment. In the high-risk environment the fitness value is sharply reduced by injury whereas the decrease exists but not very crucial for survival in the low-risk environment. Earning is far much larger in highly risky prey than the one in low risky prey. To make difference between two preys in the artificial simulation world, we make risky prey turn its head faster than the easy prey.

Simulation has been conducted with 2-D based robot simulator Enki (http://home.gna.org/enki/) of EPuck mobile robots (http://www.e-puck.org/). Each experimental group has been tested five times for each on Ubuntu 10.04 with C++ programming language. 100 seconds was given for 10 robots to hunt the prey robot within 150x150 size of artificial space.

5   Results and Analysis

We have analyzed the behavioral strategies, social structures and characteristics of leadership in evolving robot colony. At first, Fig.5 shows the changes in maximum,

average, and minimum value of vitality for difficult and easy assignments, respectively. The vitality waved with short period in the low-risk environment whereas a leader and group existed for relatively long time in the high-risk setting. A few powerful leaders emerge in the difficult situation rather than the safe environment. This result was due to the level of competition. In the high-risk environment, competition between robots was not harsh so less competitors emerged who can potentially threaten the position of the current leader.

Fig. 6 shows the developmental aspect in the ratio of social position in the robot colony. As seen in Fig.6 it showed a chaotic fluctuation until the generation 2000 in both settings. This result is attributable to the time taken for the emergence of successful hunting strategy. From the middle stage, the difference in social structure between two settings is clearly come out. Multiple groups formed frequently by multiple leaders in the low-risk environment. In the high-risk environment, however, a group maintained with many loyal followers. Similarly, the number of strangers which independently behave is larger in the first experimental group.

(a) Low-risk environment   (b) High-risk environment
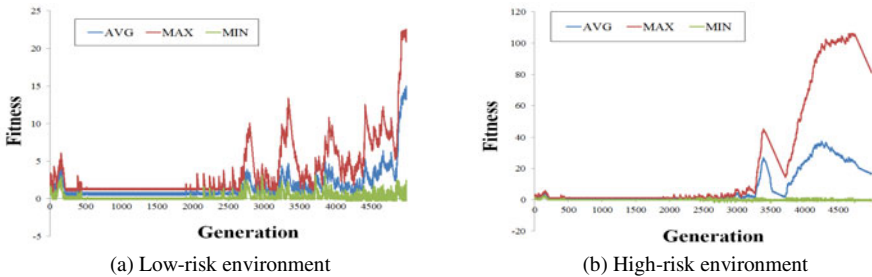
Fig. 5. Changes in average vitality of robot colony for each experimental group

(a) Low-risk environment   (b) High-risk environment
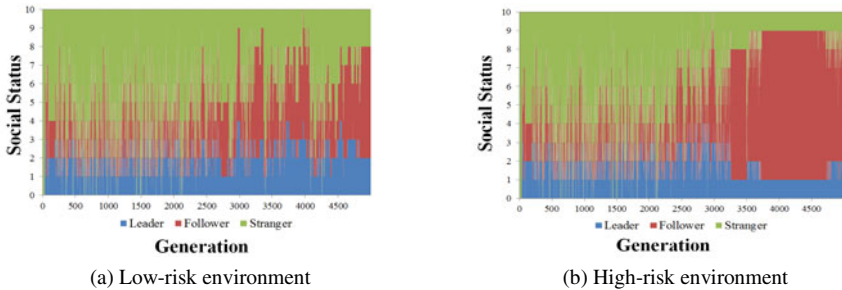
Fig. 6. Changes in the social position of robot colony

(a) Low-risk environment	(b) High-risk environment

Fig. 7. Moving patterns of robot colony

Result of the actual moving pattern depicted in Fig. 7 is also coincident with the previous results. Robots tended to show active and straightforward movement in the low-risk environment while passive behavior patterns came out in the high-risk setting. It seems that leaders gave influence on followers in a way to suppress. These results imply that harsh condition can lead more powerful form of leadership. The more robots decided to follow a heroic leader rather than facing high risk. On the other hand, less powerful but more number of leadership emerges in the lowly risk condition. Observation of the more number of independent behaviors supports this implication.

6   Conclusion and Future Works

We have analyzed the leadership behavior using evolutionary robot colony. Neither leadership nor evolutionary computation is new research area, but it seems certain that leadership in the evolving robot colony is attractive and interesting topic not touched so much, that can be connected to the more research on robot sociology. In this paper, we have designed the artificial robot colony. The robots are classified into three social position based on their own choice, and their behaviors are controlled evolutionary neural network.

For the short-term future work, we are trying to bring out simulation-level work into the reality. By utilizing a tracking system of mobile robots we are building, we can analyze the behavioral pattern with real mobile robots. Along with this we will work how robot can guess a neural network structure or probability model of other robots when their physical observation values are given.

References

1. Brownin, B., Tryzelaar, E.: Ubersim: A multi-robot simulator for robot soccer. In: International Joint Conference on Autonomous Agents and Multiagent System, pp. 948–949 (2003)
2. Floreano, D., Mitri, S., Magnenat, S., Keller, L.: Evolutionary conditions for the emergence of communication in robots. Current Biology 17, 514–519 (2007)

3. Mitri, S.: The evolution of information suppression in communicating robots with conflicting interests. PNAS 106, 15786–15790 (2009)
4. Krause, J., Hoare, D., Hemerlrijk, C.K., Rubenstein, D.I.: Leadership in fish shoals. Fish and Fisheries 1, 82–89 (2000)
5. Robbins, S., Judge, T.: Organizational behavior. Pearson 13, 356–367 (2009)
6. Dyer, J., Johansson, A., Helbing, D., Couzin, I.D., Krause, J.: Leadership, consensus decision making and collective behavior in humans. Philosophical Transactions of the Royal Society B: Biological Sciences 364, 781–789 (2009)
7. Couzin, I., Couzin, I.D., Krause, J., Franks, N.R., Levin, S.A.: Effective leadership and decision-making in animal groups on the move. Nature 433, 513–516 (2005)
8. Nabet, B., Leonard, N.E., Couzin, I.D., Levin, S.A.: Leadership in animal group motion: a bifurcation analysis. In: Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems (2006)
9. Knoester, D.B., McKinley, P.K., Ofria, C.A.: Using group selection to evolve leadership in populations of self-replicating digital organisms. In: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, pp. 293–300 (2007)
10. Goldberg, D.E.: Genetic algorithms in search, optimization, and machine learning. Addison-Wesley Publishing Company (1989)

Emergence of Purposive and Grounded Communication through Reinforcement Learning

Katsunari Shibata and Kazuki Sasahara

Dept. of Electrical & Electronic Engineering, Oita University,
700 Dannoharu, Oita 870-1192, Japan
shibata@oita-u.ac.jp

Abstract. Communication is not just the manipulation of words, but needs to decide what is communicated considering the surrounding situations and to understand the communicated signals considering how to reflect it on the actions. In this paper, aiming to the emergence of purposive and grounded communication, communication is seamlessly involved in the entire process consisted of one neural network, and no special learning for communication but reinforcement learning is used to train it. A real robot control task was done in which a transmitter agent generates two sounds from 1,785 camera image signals of the robot field, and a receiver agent controls the robot according to the received sounds. After learning, appropriate communication was established to lead the robot to the goal. It was found that, for the learning, the experience of controlling the robot by the transmitter is useful, and the correlation between the communication signals and robot motion is important.

Keywords: emergence of communication, grounded communication, reinforcement learning, neural network, robot control task.

1 Introduction

Many speaking robots have appeared recently, and interactive talking can be seen in some of them. A robot talking with humans looks intelligent at a glance, but a long interaction with them makes us notice that the partner is not a real life but a robot. One major reason must be that the communication is not grounded, but is just the manipulation of words based on pre-designed rules. Many attempts have been made to solve the "Symbol Grounded Problem"[1] for a long time. In the model of lexicon emergence in [2] or [3], extracted features of a presented object are associated with words or codes. Under the assumption of common observation between two agents, the models have a way of getting the listener's words closer to the speaker's.

They suppose patterns and symbols separately, and focus on bridging between them through specialized learning that is independent of the other learning. Steels himself said in [3], "The experiments discussed in this article all assume that agents are able to play language games, but how do the games themselves

emerge?" The question gets the heart of the problem. Primitive communication observed in animals or ancient people seems purposive such as telling food location or coming dangers. Communication should emerge in the learning in daily life, and the communication learning should not be isolated from the other learning. It is worth noting that, when we see the section of the brain, the language areas are not isolated from the other areas, nor look so different from them. The communication is not generated only by the language areas of the brain, but is generated by the whole brain as a massively parallel and flexible processing system. That enables us to consider many things simultaneously in parallel and to decide flexibly and instantly what we talk, the authors think.

The emergence of purposive communication has been aimed by evolutional approach[4] or reinforcement learning[5]. The author's group has also investigated it through reinforcement learning[6][7][8]. Discretization of the communication signal through reinforcement learning in a noisy environment was also shown[8]. However, in these cases, the environment is very simple, and learning is performed only on computer simulation.

In this paper, using a real camera, speaker, microphone, and robot, a transmitter learns to output two sounds with appropriate frequencies from more than one thousand color image signals from the camera, and a receiver learns to output appropriate motion commands from the received sounds. Each agent uses a neural network to compute the output, and learns it by reinforcement learning only from a reward when the robot reaches a goal state and a small punishment when it is close to a wall. The emergence of symbol is left as a future problem.

There are some communication robots with one or two cameras[9][10][11], but the camera is used for the perception of communication partners or environment or for giving the feeling of being gazed to the partner. The camera image is not reflected to the communication directly, and no organic integration of the camera image and communications can be seen in them.

2   Reinforcement Learning with a Neural Network

Reinforcement learning is autonomous and purposive learning based on trial and errors, and a neural network (NN) is usually used as a non-linear function approximator to avoid the state explosion due to the curse of dimensionality. An author has claimed that by the combination, parallel processing that enables to consider many things simultaneously is learned purposively, seamlessly and in harmony, and as a result, necessary functions such as recognition, memory (when using RNN) emerges to get rewards and to avoid punishments. The flexible and parallel processing is expected to contribute to saying goodbye to the "Functional Modules" approach, in which each functional module is sophisticatedly programed independently and the modules are integrated to develop an intelligent robot. It is also expected to contribute to solving the "Frame Problem".

The system is consisted of one NN whose inputs are sensor signals and whose outputs are actuator commands. Based on reinforcement learning algorithm, training signals are generated autonomously, and supervised learning is applied

using them. This eliminates the need to supply training signals from outside. In this paper, for a continuous input-output mapping, actor-critic[13] is used as a reinforcement learning method. Therefore, the outputs of the NN are divided into a critic output P and actor outputs $\mathbf{a}$. The actor output vector $\mathbf{a}$ is used as motion commands to its actuators after adding a random number vector $\mathbf{rnd}$ as an exploration factor. For learning, TD-error is represented as

$$\hat{r}_{t-1} = r_t + \gamma P(\mathbf{s}_t) - P(\mathbf{s}_{t-1}) \tag{1}$$

where r_t is the reward given at time t, γ is a discount factor, $\mathbf{s}_t$ is the sensor signal vector that is the input of the NN at time t, and $P(\mathbf{s}_t)$ is the critic output when $\mathbf{s}_t$ is the input of the network. The training signal for the critic output is computed as

$$P_{d,t-1} = P(\mathbf{s}_{t-1}) + \hat{r}_{t-1} = r_t + \gamma P(\mathbf{s}_t), \tag{2}$$

and the training signal for the actor output is computed as

$$\mathbf{a}_{d,t-1} = \mathbf{a}(\mathbf{s}_{t-1}) + \hat{r}_{t-1}\mathbf{rnd}_{t-1} \tag{3}$$

where $\mathbf{a}(\mathbf{s}_{t-1})$ is the actor output when $\mathbf{s}_{t-1}$ is the input of the NN, and $\mathbf{rnd}_{t-1}$ is the random number vector that was added to $\mathbf{a}(\mathbf{s}_{t-1})$. Then $P_{d,t-1}$ and $\mathbf{a}_{d,t-1}$ are used as training signals, and the NN with the input $\mathbf{s}_{t-1}$ is trained once according to Error Back Propagation[14]. Here, the sigmoid function whose value ranges from -0.5 to 0.5 is used. Therefore, to adjust the value range of the neural network output to that of the actual critic value, 0.5 is added to the critic output of the neural network in Eq. (1), and 0.5 is subtracted from the derived training signal in Eq. (2). The learning is very simple and general, and as you notice, no special learning for communication or the task is applied.

3 Learning of Purposive and Grounded Communication

3.1 System Architecture and Robot Control Task

Fig. 1 shows the system architecture and performed task. There are a mobile robot (e-puck) in a $30cm \times 30cm$ square field and two communication agents; a transmitter and a receiver. The transmitter has a camera that is fixed and looking down the field from above. It has a neural network (NN), and its input vector $\mathbf{s}$ is the RGB pixel values of the camera image. It also has a speaker and transmits two sounds. The frequencies of two sounds are decided by the sum of the actor output vector $\mathbf{a}$ and an exploration factor $\mathbf{rnd}$ through the linear transformation of each element to the range between 1,000Hz and 1,300Hz. The two sounds are one-second sin-waves, and come out successively with a small interval. Due to a bug in the program, the frequency of the transmitted signal was actually about 20Hz smaller than intended. The receiver has a microphone and catches the two sounds from the transmitter. The receiver also has a NN. Its input vector $\mathbf{s}$ has 60 elements, each of which represents the average spectrum over 10Hz width around its responsible frequency of one of the two sounds and is
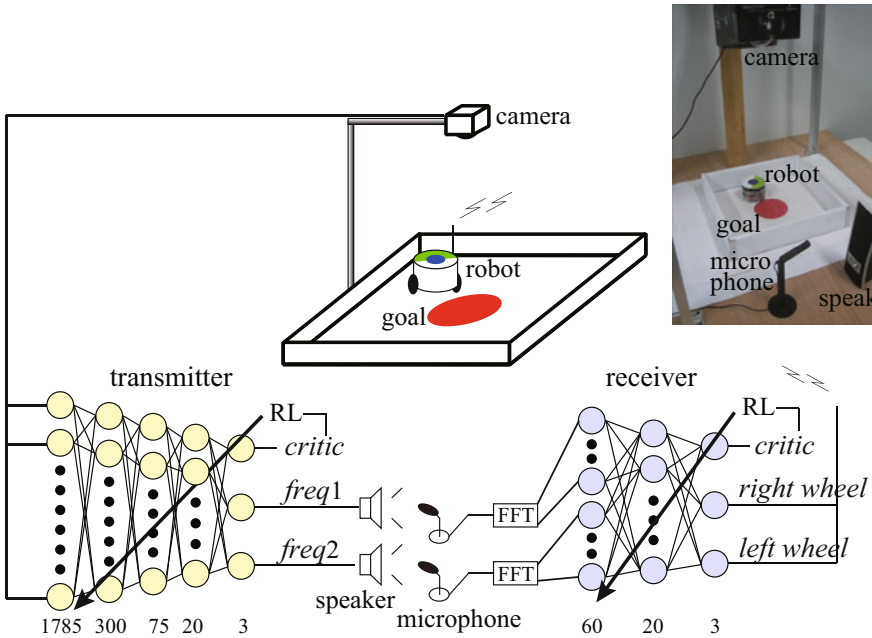
Fig. 1. System architecture and robot control task. In this figure, two speakers and two microphones are drawn, but actually, two sounds come out from one speaker with a small interval and are received by one microphone.

normalized by the maximum value. The receiver generates the control commands for the left and right wheels of the robot in proportion to the sum of its actor output vector **a** and an exploration factor **rnd**, and sends them to the robot through bluetooth.

Learning is very easy, and just proceeds according to the regular reinforcement learning independently in each agent as described in the last section. There is a big red circle in the center of the robot exploration field. When the robot center reaches the circle, the both agents get a reward 0.9 and the episode terminates. When the robot comes close to the wall, it is brought back to the position at the previous time step, and a small punishment -0.01 is imposed.

A sample raw camera image is shown in Fig. 2(a). To reduce the computational time, the image is resized to 26×20. Fig. 3 shows the definition of forward and backward and also relative and absolute orientation of the robot. The green part indicates the front of the robot, and absolute angle θ is the angle from the vertical axis of the image, and relative angle α is the angle from the line connecting to the center of the goal.

In the preliminary learning in which the NN with the input of 26×20 pixels is trained to output the relative distance and orientation ($cos\alpha, sin\alpha$) for a variety of robot locations by supervised learning, the error for the orientation outputs did not decrease so much. It would be difficult to recognize the relative orientation

(a) Sample camera image   (b) Robot-centered image

Fig. 2. Robot-centered image

Fig. 3. The definition of forward and backward, and absolute and relative orientation θ and α of the robot
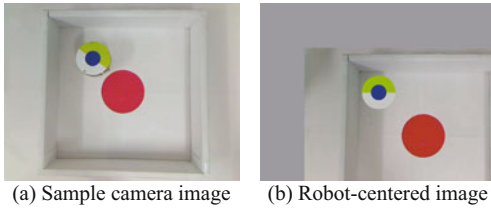
for every robot location from the image inputs. Therefore, the robot-centered image as shown in Fig. 2(b) was introduced. From the viewpoint of autonomous and seamless learning, acquisition of appropriate image shift by camera motion through learning is expected, but here, for simplicity, the image shift was given. The empty area that appears by the shift is filled with gray color as in Fig. 2(b). Furthermore, to increase the precision, the resolution of the 5×5 area around the center of the image is doubled. Each pixel color is represented by the three signals for RGB, and 1,785 signals are the input of the NN in total. Each signal is linearly normalized from -0.5 to 0.5 prior to the input.

3.2 Effect of Preparation Learning

In this task, the robot can reach the goal area by going forward or backward after changing its orientation by rotating motions. The rotational direction can be left or right, but for eliminating wasted motion, the optimal one is right for $\alpha \leq 90°$ or $180° < \alpha \leq 270°$, and left for otherwise. Around $\alpha = 90°$ or $\alpha = 270°$, the optimal direction changes drastically by the small difference of α. After learning, the robot could reach the goal successfully. However, the rotational direction was not optimal, but was always the same. That would be because, for the transmitter, the communication signals do not directly influence the robot motion, but indirectly influence it through the receiver.

Then, before the communication learning, the transmitter learns directly to control the robot by reinforcement learning as a single agent learning. After that, using the internal representation of the NN, in other words, after resetting all the connection weights between hidden and output layers to 0.0, it learns communication signals with the receiver. After the single agent learning, the rotational direction was appropriately chosen depending on the relative orientation α. Also after the following communication learning, the direction was appropriately chosen as shown in the next section. It is interesting that the previous experiments are useful for learning of appropriate communication.

3.3 Correlation between Communication Signals and Motions

One of the reasons of unsuccessful learning found during investigation is little correlation between communication signals and motions. In the receiver's NN,
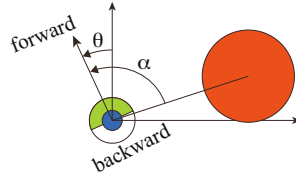
initial weights	(a) random	(b) ordered
The frequency of a communication signal (*freq*) vs The output of a hidden neuron (*output*)		
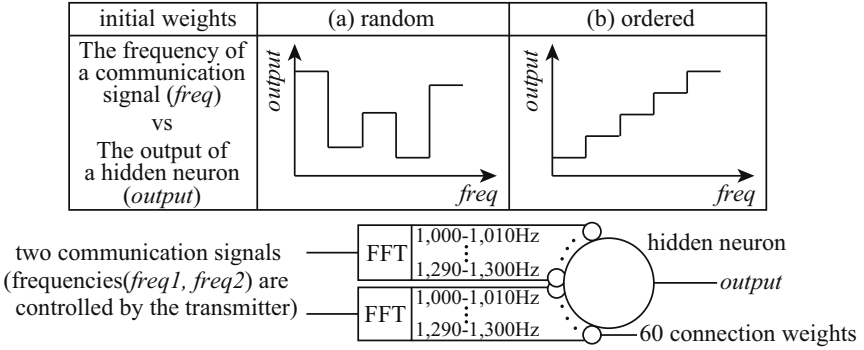
Fig. 4. The loss of the correlation between the frequency of a communication signal and the output of each hidden neuron by random initial weights in the receiver agent

each hidden neuron had a random initial connection weight to each input signal after FFT. Therefore, the output of the neuron does not change monotonically according to the frequency of a communication signal as shown in Fig. 4(a). Then, the motion commands, which are the receiver's actor output, also have little correlation with the frequency. If the correlation does not exist, it is difficult for the transmitter to know whether the frequency should be increased or decreased to make the robot motion more appropriate. Accordingly, in this research, the weights for the inputs for one communication signal to each hidden neuron increase or decrease gradually as the responsible frequency of input increases as shown in Fig. 4(b). In the same reason, the exploration factor **rnd** that is added to the receiver's actor output is ± 0.1, while the transmitter's exploration factor is ± 1.8. It is reported also in [7] that such setting is useful.

4   Experiment

Parameters in this learning are shown in Table 1. Because of the high-dimensional input, the NN in transmitter has 5 layers, while the receiver has a 3-layer NN. 6,000 episodes of learning were done. The range of initial location of the robot becomes wider gradually as the learning progresses. Fig. 5 shows two sample episodes with no exploration factors after learning. In one of the episodes (a), the robot was located upper-left area and the absolute orientation of the robot was $\theta = 0°$, that means that the green part of the robot was located upper than the white part. In the other episode (b), the robot was located lower-left area and the orientation was also $\theta = 0°$. For each episode, time series of camera image, transmitter's critic and actors (signal frequencies), and receiver's critic and actors (motion commands) are shown. In the first sample, at first, the transmitter sent a high frequency sound followed by a low frequency sound, and the robot went backward rotating anti-clockwise. After that, the transmitter sent high frequency sound and then a little high frequency sound, and the robot went backward, and finally arrived at the goal. In the second sample, at first,

Table 1. The parameters used in the learning

	transmitter	receiver
number of neurons	1785-300-75-20-3	60-20-3
learning rate	0.5	0.3
initial weight (input -> hidden)	weight after preparation learning	orderd (-2.0 - 2.0)
initial weight (hidden -> output)	random [-0.5 - 0.5]	random [-2.0 - 2.0]
exploration factor	random [-1.8 - 1.8]	random [-0.1 - 0.1]
reward	0.9	
penalty	0.01	
discount factor γ	0.96	

low-frequency sound and then high-frequency sound are sent, and the robot went forward rotating clockwise. After that, the transmitter's second sound became around the middle, and the robot went forward until it arrived at the goal.

Fig. 6(a) shows the two signal frequencies (transmitter's actor outputs) for some combinations of the robot location and absolute orientation θ. The frequencies are generated in the transmitter from the actually captured camera image. It can be seen that the frequencies are different depending on the location or orientation of the robot, but when the relative location of the goal from the robot is the same, the frequencies are similar to each other (e.g. upper left in (a-1) and lower left in (a-2)). Fig. 6(b) shows the motion commands (receiver's actor outputs) for some combinations of the two signal frequencies. To make this figure, actual sin-wave sound were emitted from the speaker, caught by the microphone, and were put into the receiver's NN after FFT. It can be seen that two motion commands change smoothly according to the two signal frequencies. Fig. 6(c) shows the relation between robot state and motion commands. The motion commands were generated from the actually captured image through the transmitter, the speaker, the microphone, FFT, and the receiver. It is shown that through appropriate communications, the robot rotated appropriately depending on the state even though the robot motion was not completely optimal.

The communication signals represent only the motions that the robot should execute, but does not represent the state or action value. Therefore, the receiver cannot represent the critic considering the robot state, but acquires the mapping from the communication signals to the robot motions. That is also shown in [15], and the problem of state confusion in the receiver was pointed in it.

5   Conclusion

It was shown that using a real mobile robot, a camera, a speaker, and a microphone, the communication from the transmitter, who saw the robot's state as the camera image, to the receiver, who generated the motion commands to the
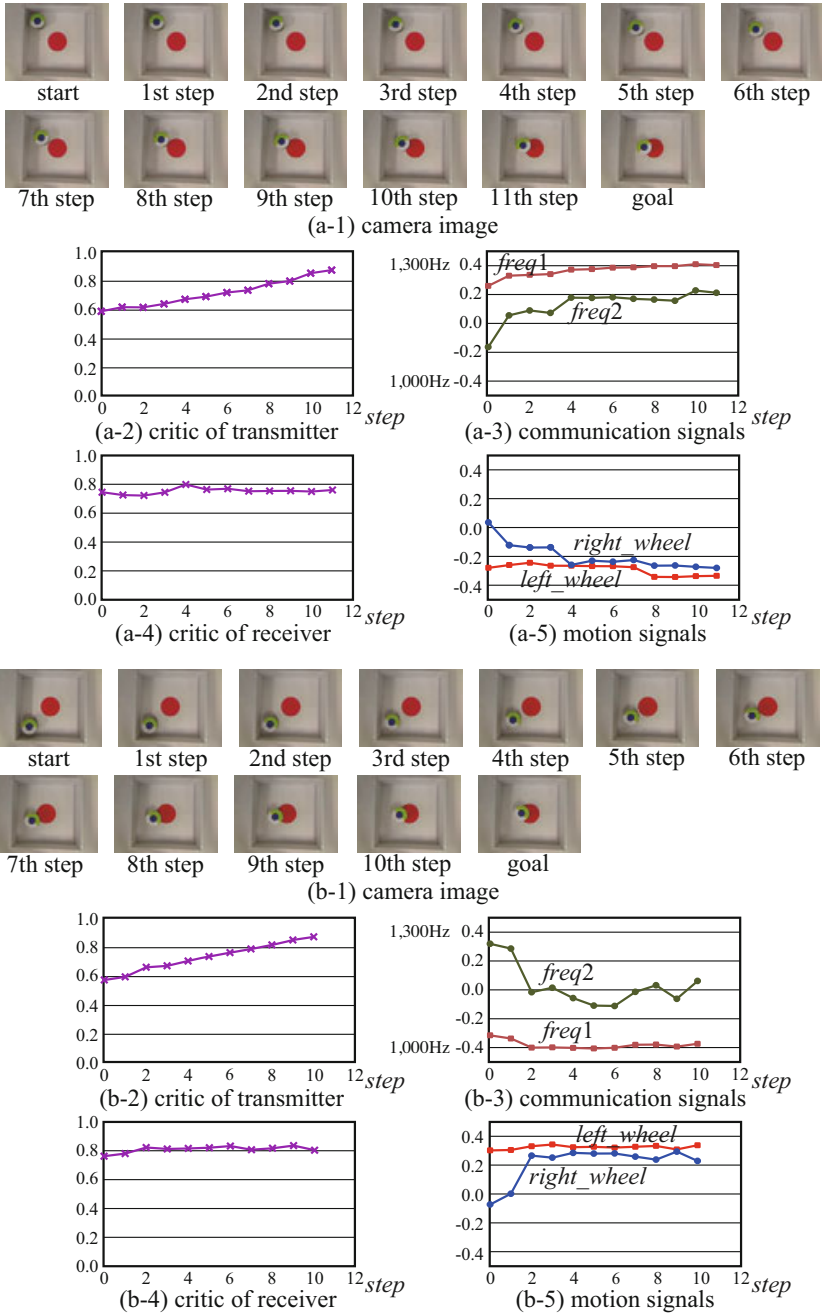
Fig. 5. The robot behavior and transmitter's and receiver's output changes in two sample episodes. Since the communication signals represent only appropriate motions and no value of state or action, the critic output does not increase in the receiver.
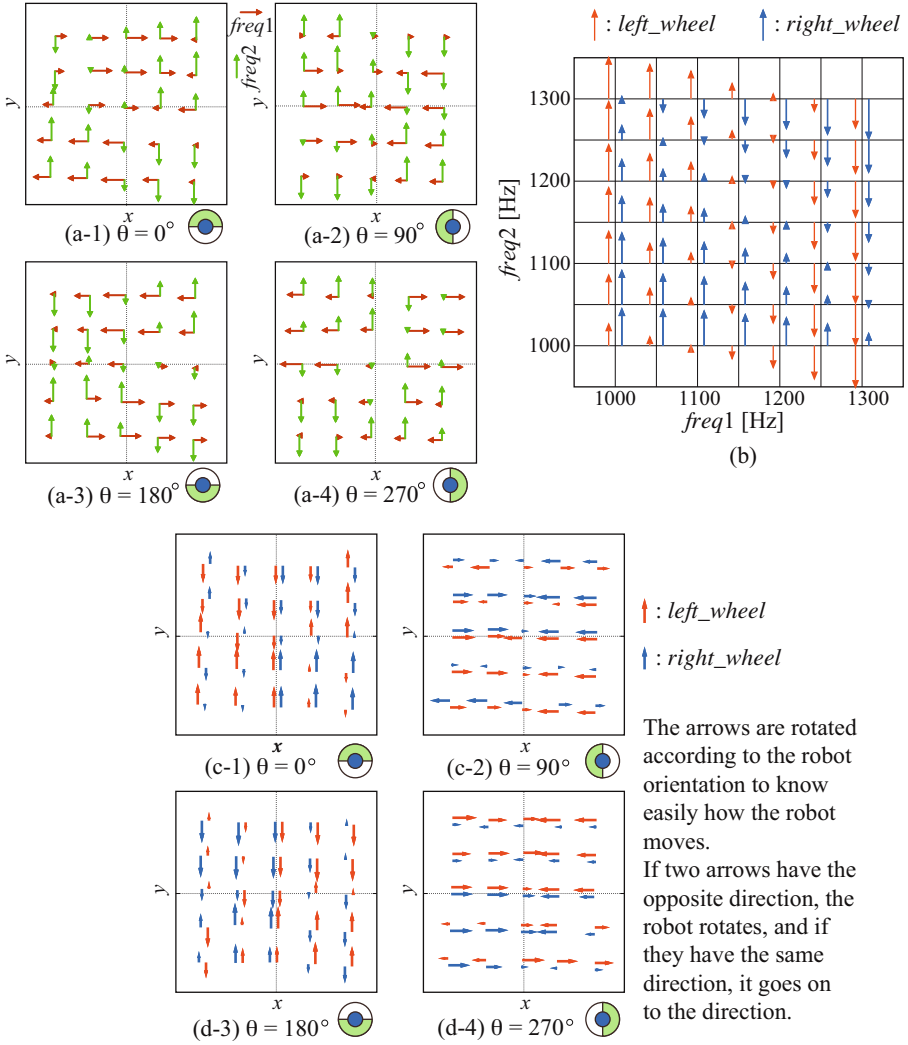
Fig. 6. (a)The frequency of communication signals ($freq1, freq2$) (transmitter's actor outputs) for some robot locations (x, y). The position of the arrows indicates the robot location on the field. The robot orientation θ is different among (a-1, 2, 3, 4). That is also shown in the small robot image beside each figure. The pair of horizontal brown ($freq1$) and vertical green ($freq2$) arrow lengths shows the frequencies of the two signals. (e.g. 1,000Hz: longest in the upper or right direction, 1,150Hz: length is 0, 1,300Hz: longest in the lower or left direction) (b)The motion commands ($left, right$) (receiver's actor outputs) for some combinations of two communication signals ($freq1, freq2$). (c)The motion commands ($left, right$) for some robot locations (x, y) and orientation θ pairs. The motion commands for each state is represented by a pair of red and blue arrows. The red arrows show the motion command for the left wheel, while the blue arrows show that for the right wheel.

robot, could be established through reinforcement learning only from a reward and punishment. It is also claimed that in the communication learning, actual control experience in the transmitter, and also the correlation between the transmitted communication signal and the final effect are important. In this paper, the communication signals are continuous, and in this meaning, the "Symbol Grounding Problem" has not been solved. However, purposive and grounded communication that includes what should be communicated considering the situation through many sensor signals and also how should the communication signals be reflected on motions was acquired through learning without any specialized learning for communication.

References

1. Harnad, H.: Symbol Grounding Problem. Physica D 42, 335–346 (1990)
2. Nakano, K., Sakaguchi, Y., Isotani, R., Ohmori, T.: Self-Organizing System Obtaining Communication Ability. Biological Cybernetics 58, 417–425 (1988)
3. Steels, L.: Evolving grounded communication for robots. Trends in Cognitive Science 7(7), 308–312 (2003)
4. Werner, G.M.: DyerM.G.: Evolution of Communication in Artificial Organisms. In: Proc. of Artificial Life II, pp. 1–47 (1991)
5. Ono, N., et al.: Emergent Organization of Interspecies Communication in Q-Learning Artificial Organs. In: Advances in Artificial Life, pp. 396–405 (1995)
6. Shibata, K., Ito, K.: Emergence of Communication for Negotiation By a Recurrent Neural Network. In: Proc. of ISADS 1999, pp. 294–301 (1999)
7. Nakanishi, M., Shibata, K.: Effect of Action Selection on Emergence of One-way Communication Using Q-learning. In: Proc. of 10th AROB, CD-ROM, vol. GS7-3 (2005)
8. Shibata, K.: Discretization of Series of Communication Signals in Noisy Environment by Reinforcement Learning. Adaptive and Natural Computing Algorithms, 486–489 (2005)
9. Mitsunaga, N., et al.: Robovie-IV: A Communication Robot Interacting with People Daily in an Office. In: Proc. of IROS 2006, pp. 5066–5072 (2006)
10. Suga, Y., et al.: Development of Emotional Communication Robot, WAMOBEA-3. In: Proc. of ICAM 2004, pp. 413–418 (2004)
11. Bennewitz, M., et al.: Fritz - A Humanoid Communication Robot. In: Proc. of ROMAN 2007, pp. 1072–1077 (2007)
12. Shibata, K.: Emergence of Intelligence through Reinforcement Learning with a Neural Network. In: Advances in Reinforcement Learning, in Tech, pp. 99–120 (2011)
13. Barto, A.G., et al.: Neuronlike Adaptive Elements That can Solve Difficult Learning Control Problems'. IEEE Trans. of SMC 13, 835–846 (1983)
14. Rumelhart, D.E., et al.: Learning Internal Representation by Error Propagation. Parallel Distributed Processing (1986)
15. Nakanishi, M., et al.: Occurrence of State Confusion in the Learning of Communication Using Q-leaning. In: Proc. of 9th AROB, vol. 2, pp. 663–666 (2004)

An Action Selection Method
Based on Estimation of Other's Intention
in Time-Varying Multi-agent Environments

Kunikazu Kobayashi, Ryu Kanehira, Takashi Kuremoto, and Masanao Obayashi

Yamaguchi University, Tokiwadai 2-16-1, Ube, Yamaguchi 755-8611, Japan
{koba,wu,m.obayas}@yamaguchi-u.ac.jp
http://www.nn.csse.yamaguchi-u.ac.jp/k/

Abstract. An action selection method based on the estimation of other's intention is proposed to treat with time-varying multi-agent environments. Firstly, the estimation level of other's intention is stratified as active, passive and thoughtful levels. Secondly, three estimation levels are formulated by a policy estimation method. Thirdly, a new action selection method by switching three estimation levels is proposed to cope with time-varying environments. Fourthly, the estimation methods of other's intention are applied to the Q-learning method. Finally, through computer simulations using pursuit problems, the performance of the estimation methods are investigated. As a result, it is shown that the proposed method can select the appropriate estimation level in time-varying environments.

Keywords: Multi-agent system, Reinforcement learning, Intention estimation, Action selection, Pursuit problem .

1 Introduction

Multi-agent systems can emerge intellectual behavior such as cooperative behavior toward a goal of agent group through mutual interaction among individual agents. In general, multi-agent systems can cope with intractable problems that single-agent systems cannot solve and dynamical environments [1]. As giving agents a reinforcement learning function, multi-agent systems can maximize its potential abilities such as cooperativeness and robustness [2,3].

To realize cooperative behavior in multi-agent systems, if agents are able to communicate with others using some kind of communication tool, agents can pick up on other's intention. Agents however have to estimate the other's intention if agents are unable to communicate with others by restrictions of robot hardware and external environments. In the present paper, we assume intention as agent's behavior with a goal and a plan after Bratman's definition [4]. In this situation, agents are required to accurately estimate the other's intention and to cooperatively act toward a goal of agent group.

Nagayuki et al. presented a policy estimation method which can estimate the other's action to be taken based on the observed information about the other's action sequence [5,6]. They successfully applied it to the Q-learning method [7] which is one of reinforcement learning methods and showed to get effective the other's policy. Meanwhile,

Yokoyama et al. proposed an approach to model action decision based on the other's intention according to an atypical situation such as human-machine interaction [8,9]. They defined three estimation levels of the other's intention and presented a computational model of action decision process to solve cooperative tasks through a psychological approach.

Although the approach of Nagayuki et al. assumes the policy estimation as the other's action prediction, they don't consider a deep intention estimation at all, i.e. a self-action prediction by others. The self-action therefore consists of a self-experience and the other's action prediction. On the other hand, The approach of Yokoyama et al. estimates the other's intention but has to learn in advance by classifying action probabilities according to goals and cannot cope with time-varying environments.

In the present paper, we propose an action selection method based on the estimation of the other's intention to treat with time-varying multi-agent environments. In Section 2, we briefly outline the Q-learning method. In Section 3, we give three estimation levels of the other's intention based on the work of Yokoyama et al. and formulate these three estimation levels using the policy estimation method of Nagayuki et al. We furthermore propose a new action selection method by switching the three estimation levels to cope with time-varying environments. At the same time, all the estimation methods are applied to the Q-learning method. In Section 4, we investigate the performance of the estimation methods through computer simulations using pursuit problems. As a result, we confirm that the proposed method can select the appropriate estimation level in time-varying environments.

2 Reinforcement Learning

Reinforcement learning is a machine learning technique that a decision-making agent takes actions and then receives rewards in an environment, and finally acquires the optimum policy by trial and error [2,3].

The Q-learning method by Watkins et al. is a representative reinforcement learning technique and guarantees that a value function will converge to the optimal solution by appropriately adjusting a learning rate in Markov decision process environments [7]. A state-action value function $Q(s, a)$ is updated by (1) so as to take the optimal action by exploring it in a learning space.

$$Q(s, a) \leftarrow (1 - \alpha)Q(s, a) + \alpha \left(r + \gamma \max_{a' \in A} Q(s', a') \right), \tag{1}$$

where s' is the next state after an agent takes action a, r is a reward at the state s', A is a set of all possible actions, α is a learning rate ($0 < \alpha < 1$), γ is a discount rate ($0 \leq \gamma \leq 1$).

Probabilistically, an agent selects action a at state s according to policy $\pi(s, a)$. Throughout the present paper, we employ the Boltzmann method defined by (2) as the policy.

$$\pi(s, a) = \frac{\exp\left(\beta Q(s, a)\right)}{\displaystyle\sum_{b \in A} \exp\left(\beta Q(s, b)\right)}, \tag{2}$$

where β is a parameter to control randomness of action selection called as inverse temperature parameter. The policy $\pi(s, a)$ referes to a probability to select action a at state s.

3   Intention Estimation Levels and Their Application to Reinforcement Learning

3.1   Intention Estimation Levels

In the present paper, intention estimation refers to the estimation of an action sequence toward a goal. We formulate three estimation levels according to the depth of the intention estimation which referes to the work of Yokoyama et al. [8,9]. Then we propose a new estimation level which can switch three levels depending on the situation. Note that we abbreviate level as Lv. and intention estimation as IE.

Lv.0 IE.   We sometimes behave without awareness of others. We call this as active behavior and label Lv.0 IE. Lv.0 IE assumes an action selection mechanism which an agent approaches a self-goal without intention estimation of others.

Lv.1 IE.   We often select actions by predicting the other's actions. We call this as passive behavior and label Lv.1 IE. Lv.1 IE is an action selection mechanism by predicting the other's actions based on an other's action history.

Lv.2 IE.   We often decide actions by estimating the other's intention. We call this as thoughtful behavior and label Lv.2 IE. Lv.2 IE is an action selection mechanism not only by predicting the other's actions but also by estimating the other's intention based on the other's situation.

Lv.3 IE.   We often choose actions by changing estimation levels depending on the situation. We label this Lv.3 IE. Lv.3 IE is an approach to switch the above three estimation levels, i.e. Lv.0, Lv.1, and Lv.2 IEs depending on the situation.

In the next section, we implement these estimation levels with reinforcement learning.

3.2   Application to Reinforcement Learning

At first, we formulate the estimation levels described in 3.1 in order to apply them to reinforcement learning. In the present paper, we employ the Q-learning method described in 2 as a reinforcement learning method. After that, we propose a new reinforcement learning system which can switch three estimation levels depending on the situation.

Action Selection Method at Lv.0 IE.   Lv.0 IE realizes active action selection without considering the other's intention. The learning at Lv.0 IE therefore employs the standard Q-learning method.

To begin with, let us denote a self-state as s_s, a self-action as a_s ($\in A_s$), and an other's action as a_o ($\in A_o$). Note that A_s and A_o refer to the sets of all possible actions by self and the other, respectively. In the present paper, both action elements of self and those of the other assume completely identical, i.e. $A_s = A_o$. Let us denote a Q-function as $Q(s_s, a_s, a_o)$. An update rule of $Q(s_s, a_s, a_o)$ is represented by

$$Q(s_s, a_s, a_o) \leftarrow (1-\alpha)Q(s_s, a_s, a_o) + \alpha \left(r + \gamma \max_{a_s' \in A_s} \bar{Q}(s_s', a_s') \right), \qquad (3)$$

where s'_s is a next self-state and $\bar{Q}(s_s, a_s)$ is the average of $Q(s_s, a_s, a_o)$ with respect to a_o.

$$\bar{Q}(s_s, a_s) = \sum_{a_o \in A_o} \frac{1}{|A_o|} Q(s_s, a_s, a_o), \tag{4}$$

where $|A|$ denotes the number of elements in set A.

We employ the Boltzmann method in (2) as action selection. Note that the Q-function in (2) should be replaced by (4). At Lv.0 IE, self-action a_s with the higher value of $\bar{Q}(s_s, a_s)$ tends to be selected.

Action Selection Method at Lv.1 IE. Lv.1 IE realizes passive action selection with predicting the other's actions. The learning at Lv.1 IE is assumed as the Q-learning method based on other's action estimation.

We employ the policy estimation method by Nagayuki et al. [5,6] for Lv.1 IE. The method predicts an other's action using a policy estimation function $P_s(s_s, a_o)$. The P-function $P_s(s_s, a_o)$ is updated by (5) for all the other's actions to be taken, i.e. $a_o (\in A_o)$

$$P_s(s_s, a_o) \leftarrow (1-\rho)P_s(s_s, a_o) + \begin{cases} \rho & (a_o = a_o^*), \\ 0 & \text{(otherwise)}, \end{cases} \tag{5}$$

where a_o^* is the actual other's action and ρ is a positive parameter $(0 \leq \rho \leq 1)$. As updating P-value by (5), P-value with a_o^* increases and the other P-values decrease. Repeatedly updating P-values, an agent can predict other's actions. Note that $\sum_{a_o \in A_o} P_s(s_s, a_o) = 1$ holds at any time.

An update rule of $Q(s_s, a_s, a_o)$ at Lv.1 is denoted by

$$Q(s_s, a_s, a_o) \leftarrow (1-\alpha)Q(s_s, a_s, a_o) + \alpha \left(r + \gamma \max_{a'_s \in A_s} \bar{Q}(s'_s, a'_s) \right), \tag{6}$$

where $\bar{Q}(s_s, a_s)$ is a weighted average of $Q(s_s, a_s, a_o)$ with respect to a_o.

$$\bar{Q}(s_s, a_s) = \sum_{a_o \in A_o} P_s(s_s, a_o) Q(s_s, a_s, a_o). \tag{7}$$

We also employ the Boltzmann method in (2) at Lv.1 IE as action selection. Note that the Q-function in (2) should be replaced by (7). As introducing the policy estimation method into Q-learning, Q-values are able to update by predicting other's actions. At Lv.1 IE, action a_s with the higher value of $\bar{Q}(s_s, a_s)$, i.e. the average of $Q(s_s, a_s, a_o)$ with respect to policy estimation function P_s tends to be selected. In this way, the prediction of other's actions reflects self-action selection. As a result, an agent can gradually predict other's actions.

Action Selection Method at Lv.2 IE. Lv.2 IE realizes thoughtful action selection which an agent decides a self-action by estimating the other's intention based on other's situation. The agent should therefore consider the self-intention which is estimated by the other.
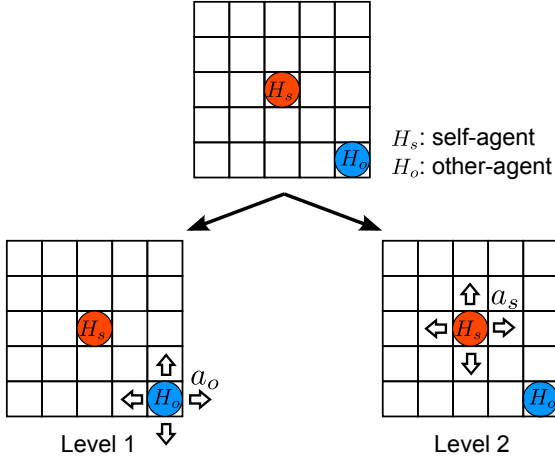
Fig. 1. Difference of policy estimation between Lv.1 and Lv.2

In the present paper, the estimation of self-intention by the other realizes by replacing the other's position with the self-position as shown in Fig. 1. The P-function is updated in the similar with Lv.1 IE. Since the P-function at Lv.2 IE is assumed as the self-policy estimation by the other, it is denoted by P_o. The P-function at state s_o is updated by (8) for all the self-actions to be taken, i.e. $a_s(\in A_s)$

$$P_o(s_s, a_s) \leftarrow (1 - \rho)P_o(s_s, a_s) + \begin{cases} \rho & (a_s = a_s^*), \\ 0 & \text{(otherwise)}, \end{cases} \tag{8}$$

where a_s^* is the actual self-action. The P-value that an agent actually took increases according to (8). The agent predicts an action which the other desires for the self.

An update rule of $Q(s_s, a_s, a_o)$ using P_o at Lv.2 is denoted by

$$Q(s_s, a_s, a_o) \leftarrow (1 - \alpha)Q(s_s, a_s, a_o) + \alpha \left(r + \gamma \max_{a_s' \in A_s} \bar{Q}(s_s', a_s') \right), \tag{9}$$

where $\bar{Q}(s_s, a_s)$ is a weighted average of $Q(s_s, a_s, a_o)$ with respect to a_s.

$$\bar{Q}(s_s, a_s) = \sum_{a_o \in A_o} P_o(s_s, a_s)Q(s_s, a_s, a_o). \tag{10}$$

We also employ the Boltzmann method in (2) at Lv.2 IE as action selection. Note that the Q-function in (2) should be replaced by (10). At Lv.2 IE, self-action a_s with the higher value of $\bar{Q}(s_s, a_s)$, i.e. the average of $Q(s_s, a_s, a_o)$ with respect to policy estimation function P_o tends to be selected.

Action Selection Method at Lv.3 IE. Lv.3 IE is an approach to switch three estimation levels, i.e. Lv.0, Lv.1, and Lv.2 IEs depending on the situation.

Since an observed state will change with time in real environments, an agent has to appropriately select an action in time-varying environments. If the estimation level is fixed, however, the agent has difficulty adjusting to the environment. The agent therefore needs to appropriately change the estimation levels.

We propose a selective method of the estimation levels in (11) to cope with time-varying environments.

$$c = \arg \max_{i \in \{0,1,2\}} PQ_i, \tag{11}$$

where PQ_i $(i = 0, 1, 2)$ is defined as follows.

$$
\begin{cases}
PQ_0 = \displaystyle\sum_{a_s \in A_s} \sum_{a_o \in A_o} \frac{1}{|A_o|} Q(s_s, a_s, a_o), \\
PQ_1 = \displaystyle\sum_{a_s \in A_s} \sum_{a_o \in A_o} P_s(s_s, a_o) Q(s_s, a_s, a_o), \\
PQ_2 = \displaystyle\sum_{a_s \in A_s} \sum_{a_o \in A_o} P_o(s_s, a_s) Q(s_s, a_s, a_o).
\end{cases} \tag{12}
$$

The proposed selective method is described as follows. Firstly, we calculate PQ_i, i.e. the product sum of P-values and Q-values. Note that P-values at Lv.0 IE mean the equal probability because they don't predict actions and estimate intention. Secondly, we compare the values of PQ_i and choose the estimation level c that has the maximum value of PQ_i. Note that we use the update rules of P-values and Q-values as described before. We can therefore select an estimation level according to the learning situation of P-values and Q-values.

4 Computer Simulation

4.1 Problem Setting

A pursuit problem is a well-known multi-agent problem which plural hunters pursuit preys (or a prey) and catch them in a grid field. The followings are assumed in the present paper.

- 9×9 grid field with a torus structure in Fig. 2.
- Two hunters (H_1 and H_2) and two preys (P_1 and P_2) in the field. Initially, H_1 and H_2 are located in the center of the field, P_1 is located near from the hunters, and P_2 is located far from the hunters as shown in Fig. 2(a). It allows that hunters and preys are occupied in the same cell.
- Two hunters can observe all the cells (complete observation) and act according to their own estimation levels. The hunters can synchronously move up, down, left, or right by one cell or stay on the same cell.
- A goal state is assumed that each hunter is occupied in one of four adjacent cells. An example of the goal state is depicted in Fig. 2(b).

(a) Initial position   (b) A goal position
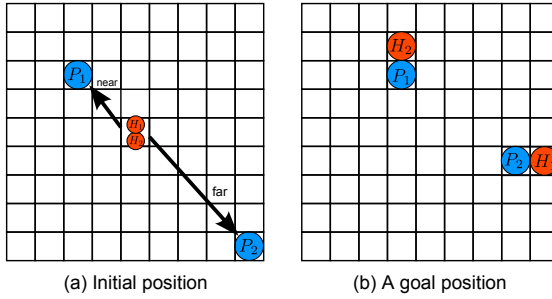
Fig. 2. (a) Initial position of two hunters (H_1 and H_2) and two preys (P_1 and P_2), (b) an example of a goal position

4.2   Simulation Setting

Two hunters get a positive reward $r = 50$ if a goal state is reached and get a negative reward $r = -0.01$ if otherwise. The number of steps is limited to 30,000 and we start a next trial if it reaches the limit.

We prepare the following two kinds of simulation setting according to behavioral patterns of two preys, P_1 and P_2.

- Simulation 1
  - P_1 can only move up.
  - P_2 can only move right.
- Simulation 2
  - Two preys can only move right before 1,500 episodes.
  - Two preys can only move left after 1,500 episodes.

Under this simulation setting, each hunter has to choose a different prey with the other hunter as a target. Since the initial positions of P_1 and P_2 are different, one hunter needs to choose P_2 as a target with considering the other hunter. In simulation 2, hunters are required to adjust to the change of the environment.

The parameters were selected as $\alpha = 0.1$, $\gamma = 0.99$, $\beta = 10$, and $\rho = 0.75$. Initial Q-values and P-values were set to 0.1 and 0.2, respectively. These parameters were selected so as to get the best performance through preliminary simulations.

4.3   Simulation Results

The learning curves for four combinations of estimation levels in Simulation 1 and 2 are shown in Figs. 3 and 4, respectively. In all the simulations, the number of steps is averaged for 10 trials. In this figure, Lv.i-j ($i, j \in \{0, 1, 2, 3\}$) refers to pairs of estimation levels which assigned for two hunters. For example, we denote Lv.0-2 if H_1 is Lv.0 IE and H_2 is Lv.2 IE. We pick up the representative pairs of estimation levels out of 10 pairs, i.e. Lv.0-0, Lv.1-1, Lv.2-2, and Lv.3-3. We also enlarge the learning curves around the last episodes, i.e. from 2,800 to 3,000 episodes for comparison. The
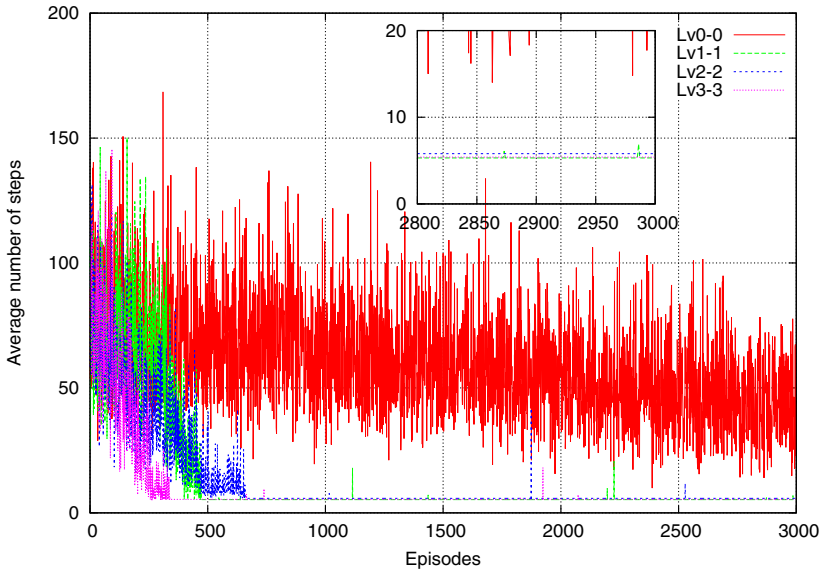
Fig. 3. Learning curves of in Simulation 1
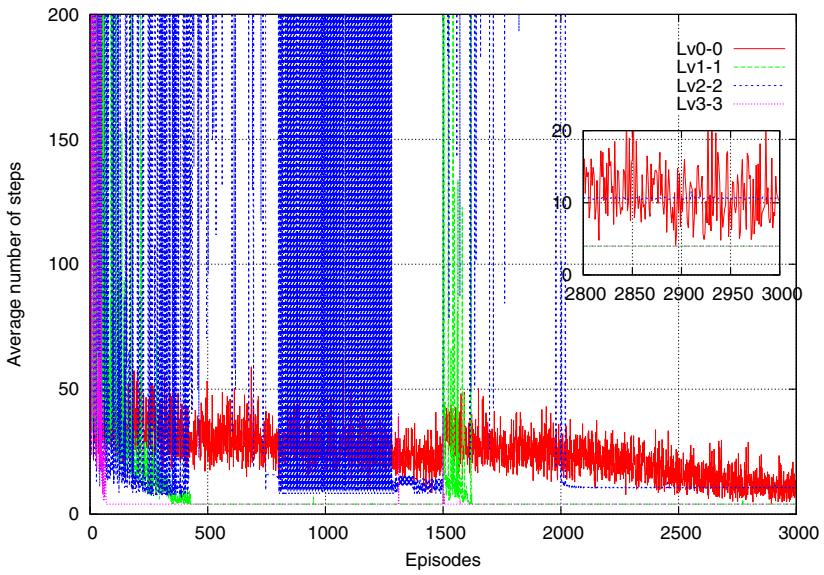
Fig. 4. Learning curves in Simulation 2

transition diagrams of estimation levels of two Lv.3's hunters in simulation 1 and 2 are shown in Figs.5 and 6, respectively. Although we need to update both P-values and Q-values, we initialized Q-values with the learned Q-values without loss of generality. We got the learned Q-values at Lv.0-0 after 3,000 episodes. In this situation, the agents don't have any advantage or disadvantage.
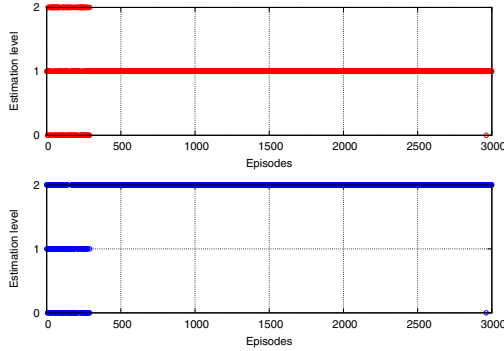
Fig. 5. Transition diagrams of estimation levels of two Lv.3's hunters in Simulation 1
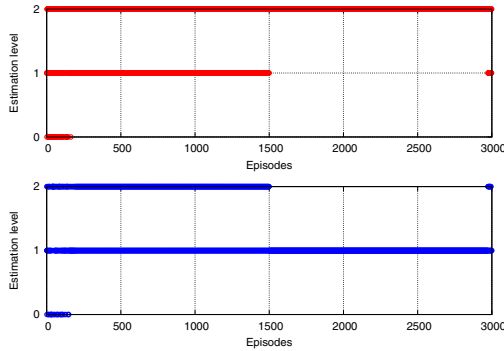
Fig. 6. Transition diagrams of estimation levels of two Lv.3's hunters in Simulation 2

As seen in Figs.3 and 4, the convergence of Lv.3-3 (combination of the proposed method) is faster than other combinations of estimation levels, i.e. Lv.0-0, Lv.1-1, and Lv.2-2. In simulation 2, as the environment is changed at 1,501 episode, the other combinations other than Lv.3-3 get increase their average number of steps. As seen in Figs.5 and 6, the estimation levels of two hunters begin at Lv.0 IE and then transit to Lv.1 IE and Lv.2 IE. Finally, H_1 and H_2 choose Lv.1 IE and Lv.2 IE in simulation 1, respectively and they select Lv.2 IE and Lv.1 IE, respectively. All the combinations of estimation levels without Lv.3 IE are six. After we conduct simulations with all the combinations, we found that Lv.1-2 showed the best performance in both simulation

1 and 2. Consequently, Lv.3-3 automatically searched the best combination, i.e. Lv.1-2. This result agrees with the work of Yokoyama et al.[8,9] That is, Yokoyama et al. pointed out that the best performance out of six combinations was Lv.1-2.

5   Summary

In the present paper, we have proposed an action selection method based on the estimation of the other's intention to treat with time-varying multi-agent environments. Firstly, we have stratified the estimation levels of the other's intention as active, passive and thoughtful levels incorporating the work of Yokoyama et al. Secondly, we have formulated three estimation levels using the work of Nagayuki et al. Thirdly, we have proposed a new action selection method by switching the three estimation levels to cope with time-varying environments. Fourthly, the estimation methods of the other's intention has been applied to the Q-learning method. Finally, through computer simulations using pursuit problems, we have investigated the performance of the estimation methods. As a result, we have confirmed that the proposed method could select the best combination of estimation levels even in time-varying environments.

References

1. Stone, P., Veloso, M.: Multiagent Systems: A Survey from a Machine Learning Perspective. Autonomous Robots 8(3), 345–383 (2000)
2. Kaelbling, L.P., Littman, M.L., Moore, A.P.: Reinforcement Learning: A Survey. Journal of Artificial Intelligence Research 4, 237–285 (1996)
3. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press (1998)
4. Bratman, M.E.: Intention, Plans and Practical Reason. Harvard University Press (1987)
5. Nagayuki, Y., Ishii, S., Ito, M., Shimohara, K., Doya, K.: A Multi-Agent Reinforcement Learning Method with the Estimation of the Other Agent's Actions. In: Proceedings of the Fifth International Symposium on Artificial Life and Robotics, vol. 1, pp. 255–259 (2000)
6. Nagayuki, Y., Ito, M.: Reinforcement Learning Method with the Inference of the Other Agent's Policy for 2-Player Stochastic Games. Transactions on the Institute of Electronics, Information and Communication Engineers J86-D-I(11), 821–829 (2003) (in Japanese)
7. Watkins, C.J.C.H., Dayan, P.: Q-learning. Machine Learning 8(3-4), 279–292 (1992)
8. Yokoyama, A., Omori, T., Ishikawa, S., Okada, H.: Modeling of Action Decision Process Based on Intention Estimation. In: Proceedings of Joint 4th International Conference on Soft Computing and Intelligent Systems and 9th International Symposium on advanced Intelligent Systems, vol. TH-F3-1 (2008)
9. Yokoyama, A., Omori, T.: Model Based Analysis of Action Decision Process in Collaborative Task Based on Intention Estimation. Transactions on the Institute of Electronics, Information and Communication Engineers J92-A(11), 734–742 (2009)

Describing Human Identity Using Attributes

Zhuoli Zhou[1], Jiajun Bu[1], Dacheng Tao[2], Luming Zhang[1],
Mingli Song[1,*], and Chun Chen[1]

[1] Zhejiang Provincial Key Laboratory of Service Robot, Computer Science College,
Zhejiang University
[2] Centre for Quantum Computation and Intelligent Systems,
University of Technology, Sydney
{zhlzhou,jjb,zglumg,brooksong,chenc}@zju.edu.cn,
dacheng.tao@gmail.com

Abstract. Smart surveillance of wide areas requires a system of mul-
tiple cameras to keep tracking people by their identities. In such multi-
view systems, the captured body figures and appearances of human, the
orientation as well as the backgrounds are usually different camera by
camera, which brings challenges to the view-invariant representation of
human towards correct identification. In order to tackle this problem, we
introduce an attribute based description of human identity in this paper.
Firstly, two groups of attributes responsible for figure and appearance
are obtained respectively. Then, Predict-Taken and Predict-Not-Taken
schemes are defined to overcome the attribute-loss problem caused by
different view of multi-cameras, and the attribute representation of hu-
man is obtained consequently. Thirdly, the human identification based
on voter-candidate scheme is carried out by taking into account of human
outside of the training data. Experimental results show that our method
is robust to view changes, attributes-loss and different backgrounds.

Keywords: Human identification, attributes learning, camera networks.

1 Introduction

Systems for recognizing human identity under multiple cameras are becoming
more and more important in a variety of research areas. They can be used for
human activity pattern analysis. In simulators, they can be used to evaluate
crowd flow or evacuation under multi-camera networks.

Previous approaches in this domain can be roughly classified into two cate-
gories: ones that extract human spatial-temporal phenomenon to characterize
the motion of an individual, e.g. gait[1]. These paradigms, which require ob-
served motion keeps invariant during the duration of the video, usually is not
robust due to camera views. The other approaches extract a feature set using
appearance for a frame, e.g. faces[2]. Face recognition has been intensively in-
vestigated for more than ten years. The state-of-the-art face recognition systems

---

* Corresponding author.

yield satisfactory performance only when confronted with controlled conditions such as invariant to illumination and pose, and high resolution images.

Smart surveillance system needs recognizing human identity even under complicated conditions, such as different viewpoints, poses changes and abnormal human motion. Moreover, different from traditional object recognition, query object (human identity) may be in a different appearance feature space from the training data which requires the system be flexible and feasible for knowledge transfer[3]. Therefore, previous techniques, which heavily relied on invariant to pose, motion, face and fixed feature spaces, face significant difficulties.

In recent years, attributes are used to describe, compare and categorize objects[4]. Attributes, which are learned from annotation, make it possible to describe objects and identify objects based on textual descriptions. This capacity to infer attributes is instinctive superior on viewpoint changes[3,4]. Moreover, when the system faces a new kind of object, attributes can still describe something even though we cannot identify it[3].

However, using conventional attribute based method to identify human faces difficulties. On one hand, using attributes to model the appearance of suspicious identity still challenging since the features obtained from different cameras are usually diverse. Most previous research only takes effect on a small degree of viewpoint variations instead of dramatic variations in the smart surveillance scenario. On the other hand, previous object recognition methods, which work under the assumption that *there always is an object*, face the challenge of distinguishing *The object is not A* and *There isn't an object* where the traditional classifier both answers "no" when classifying object A. The problem *attribute-loss*, which happened under such assumption, sometimes causes significant influence on the recognition results, e.g. observing face from back or observing bag on the back from the front. To our best knowledge, the attribute-loss problem has not been directly tackled before.

In order to tackle the above mentioned problems, we present a new attribute based model for predicting human identity by modelling human figure and appearance in video frames. First, our attributes model with spatial and view information ensures the representation of human identity keeping invariant to view-points changes. Secondly, our model provides two different schemes: *Predict-Taken* and *Predict-Not-Taken* to solve the attribute-loss problem which usually caused by different views of multiple cameras. Thirdly, with the attributes based representation, the identification process uses a voter-candidate scheme to meet the requirement of knowledge transferring on human by taking into account of human outside of the training data.

2   Our Approach

In order to build a probabilistic model towards an attribute-based, multi-view representation of human identity, the input video and query video undergo the following preprocessing steps:

- Target human modelling and body attribute learning
- Appearance attribute model learning and inference

After the above two steps, the obtained attributes representations of human identity are grouped into two sets, the voters and the candidates. For simplicity, we assume that all attributes representation have binary values such that the attribute representation $a_y = (a^1, a^2, ..., a^A)$ for any human y are fixed-length binary vectors, where A is the number of attributes used to describe target object. If the set of the voters is defined as $V = \{v_i; i = 1...n\}$, where $v_i = (a^1_{v_i}, ..., a^A_{v_i})$ and the set of candidates in a certain query video is defined as $C = \{c_i; i = 1...n\}$, where $c_j = (a^1_{c_j}, ..., a^A_{c_j})$, then the distance between voter v_i and candidate c_j is denoted by a A-dimension vector, i.e.:

$$x_{i,j} = (|a^1_{v_i} - a^1_{c_j}|, ..., |a^A_{v_i} - a^A_{c_j}|) \tag{1}$$

Let $(x_{1,1}, y_{1,1}), ..., (x_{n,m}, y_{n,m}) \in \mathcal{X} \times \mathcal{Y}$ be training sample where $\mathcal{Y} = \{0, 1\}$ indexes the relationship between voter and candidate. The training task is to learn a classifier $f : \mathcal{X} \to \mathcal{Y}$ to label any given voter-candidate pair. This scheme is flexible enough to handle different combination of attributes which don't appear in the training data.
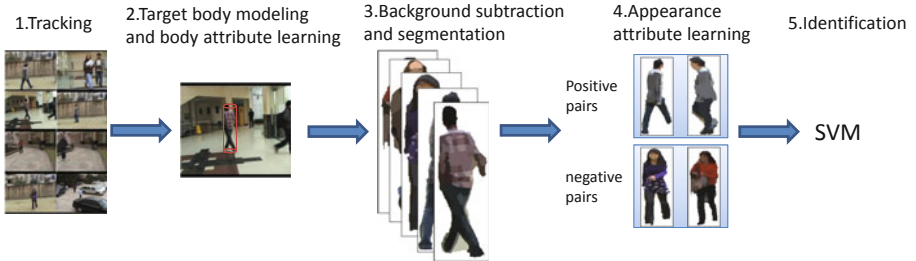
Fig. 1. The steps of human identification with input videos

2.1   Target Human Modeling and Body Attribute Learning

At first, we learn the model of 3D human body in the videos. The selection of such model should provide the following information: 1) the movement orientation, which corresponds to the body viewpoint, that is the key parameter of 3D human appearance model which will be used in the next section; 2) body figure information, such as the height and width etc., from which we can obtain attribute such as fat, tall and slow. We use a modified 3D person tracking algorithm [5] based on a Bayesian framework on 3D state space and body model to obtain such information. Given a video clip with background subtraction (in this paper, we use [6]), the human body is represented by a 7-dimensional vector:

$$U = (u, \dot{u}, v, \dot{v}, \alpha, H, W) \tag{2}$$

where (u, v) denotes the ground plane position of object, $(\dot{u}, \dot{v})$ denotes the speed of the object, α denotes the movement orientation in 8 discrete directions $\{0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°\}$, and (H, W) denotes the height and width of the object, i.e.,

$$p(W_k|W_{k-1}, \alpha_{k-1}) = p(W_k|W_{k-1})p(W_k|\alpha_{k-1}) \tag{3}$$

where the first term on the right hand imposes some temporal continuity to the estimated width, and is simply modelled as constant model(i.e. $p(W_k|W_{k-1}) = \mathcal{N}(W_k; W_{k-1}, \sigma_W^2)$), where $\mathcal{N}$ is a Gaussian distribution) and the second term defines as $p(W_k|\alpha_{k-1}) = \mathcal{N}(W_k; W_0^{\alpha_{k-1}}, \sigma_{W_0})$ where we choose $W_0^{\alpha_{k-1}}$ as a uniform prior for different orientation.

The above model can automaticlly select a frame that has the lowest speed during the duration without view changes. This selection is based on such observation that higher speed causes *Virtual Shadow* which may add extra information unexpectedly. Then, the foreground of the selected frame is normalized to a still image for appearance attribute learning.

2.2   Appearance Attribute Model

In this section, we introduce a latent-topic bag-of-visual word model with view and spatial information(vs-LTM) for describing 3D human appearance attributes. Most of the previous work in human identity recognition has taken the approach of assuming a small variation of rotation. We argue that using the view parameter learned in the previous section, our multi-view attribute representation can provide a more accurate estimation of human identity in application. Unlike 2D attribute model[3], however, part of attributes of 3D body may not be observed in 2D surveillance screen. Our goal is, therefore, not only to learn the observed appearance attributes but also to estimate the unseen attributes to solve the attribute-loss problem.

The spatial correlation of attributes has not been considered in previous work. However, earlier research infer that the pixels in the region are similar with respect to some appearance feature, such as intensity, color, or texture[7]. To explicitly model the spatially coherent nature of images, we assume the visual words in the same Homogeneous Region(HR)[7] share the same attribute. Thus, we average the pixels in a patch as the color feature and use Mean-Shift with small spatial bandwidth to over-segment the image into several regions using such color feature. Then, we detailed the model learning and inference as followings.

Model Learning. Given an image I, a set of N patches are extracted. For each patch, the model observes its position X_i and visual word assignment Y_i. Then, with the partitioned region $r = 1, 2, ..., R$ and the relationship of r and X_i, the process that generates the image I formally from the model is as follows:

1. A view-point T is modeled by a multinomial distribution $T \sim Mult(\alpha)$ where α is obtained as described in Sect 2.1 $\alpha \sim p(\alpha_k|\dot{u}, \dot{v}, \alpha_{k-1})$.

2. An image has a list of attributes, denoted by **a**. For a region r, an attributes a_j is sampled from a uniform distribution $a_j^r \sim 1/A$, where A is the length of attribute list.
3. For a topic k of the material value, a multinomial parameter ϕ_k is sampled from Dirichlet prior $\phi_k \sim Dir(\beta)$.
4. For material value of an attribute j, a multinomial parameter π_j over K topics is sampled from Dirichlet prior $\pi_j \sim Dir(\eta)$.
5. A topic label z_n of word n is sampled from the discrete distribution of attribute a_j^r, $z_{ji} \sim Discrete(\pi_j)$.
6. The value Y_n of word n is sampled from the discrete distribution of topic z_{ji}, $Y_{ji}^r \sim Discrete(\phi_k)$

The joint distribution of Y^i of attribute j selected from the total of A attributes, given an image I and a region i, can be written as:

$$P_j(Y^r|\alpha, \beta, \eta, K, A) = \prod_i^{N^r} \prod_k^K p_j(T|\alpha) P_j(\pi_j|\eta) P_j(z_{ji}|\pi_j) P_j(Y_i^r|\pi_j, \beta) \quad (4)$$

For the attribute learning task, a region is classified into attribute j^* if

$$j^* = \arg \max_{1 \leq j \leq A} \pi(j) \quad (5)$$

For the image I, the attributes value a_j can be written as the normalized result over all the regions in this image:

$$\hat{a}_j = \frac{a_j}{\sum_n a_n} \quad (6)$$

Model Inference. z_{jn} can be sampled through a Gibbs sampling procedure integrating out π_j and ϕ_k, we compute the material value at first:

$$p(z_{ji} = k|\mathbf{z_{-ji}}, \mathbf{Y}, \eta, \beta) \propto \frac{n_{-ji,y_{ji}}^{(k)} + \beta_{y_{ji}}}{\sum_y (n_{-ji}^{(k)} + \beta_y)} \cdot \frac{n_{-ji,k}^{(j)} + \eta_k}{\sum_{k'} (n_{-ji,k'}^{(j)} + \eta_{k'})} \quad (7)$$

where $n_{-ji}^{(k)}$ is the number of words in the corpus with value y assigned to topic k excluding word i in attribute j, and $n_{-ji,k'}^{(j)}$ is the number of words in attribute j assigned to topic k excluding word i in attribute j. Eq 7 is the product of two ratios: the probability of word y_{ji} under topic k and the probability of topic k in attribute j.

With a set of samples from the posterior distribution $p(z|y)$, statistics that are independent of the content of individual topics can be computed by integrating across the full set of samples. For any single sample we can estimate ϕ_k and π_j from the value **z** by

$$\phi_{ik}^y = \frac{n_i^{(y)} + \beta}{n_i^{(\cdot)} + W\beta} \quad \pi_{ij}^y = \frac{n_i^{(\alpha_j)} + \eta}{n_i^{(\cdot)} + T\eta} \quad (8)$$

2.3 Attribute-Loss

As we discussed above, the changes of view usually leads to attribute-loss. The attribute-loss can be defined as follows:if attribute a_j is labeled "yes" on view θ, and "no" on view δ, attribute a_j is lost on view-point δ. Before attributes learning, we use two different schemes: *Predict-Taken* and *Predict-Not-Taken*. The former scheme is based on such situation that even though the attribute is not observed due to view, the target human is prone to be the one such as *black hair* and *yellow face* and we predict this attribute value "yes" before learning; meanwhile the latter one treats this attribute is "no" if the attribute misses hit in the next learning step which actually ignores attribute-loss problem and believe that non-observed attribute do not exist. Furthermore, for different attributes, we take different schemes based on the rate, which is averaged by ten labelling results, of attribute-loss happened through a threshold value.

3 Experiment

3.1 Data Set

Data Organization. In this experiment, we use videos captured by five different cameras in different locations of the campus. This data set includes 40 short videos for training from 5 human models in 8 orientations, each of which is about 10 seconds, and 10 long videos for testing, each of which is about 20 minutes. Specially, locations are chosen to ensure that the human appears in two cameras at least. All the background subtraction results are resized such that the height has 300 pixels and the weight has 100 pixels. We use 29 attributes to describe these human, which are divided into two groups: body figure attribute such as *fat, thin, tall, low, fast, slow* and appearance attribute such as *black hair, yellow face, white jacket, gray trousers, black bag* etc.. There are totally 6 body figure attributes and 23 appearance attributes. The orientation-human-attribute relationship is labeled by human subjects and presented in a $8 \times 10 \times 29$ matrix $\mathcal{M}$, Fig 2(a) illustrates a subset of this matrix from in orientation
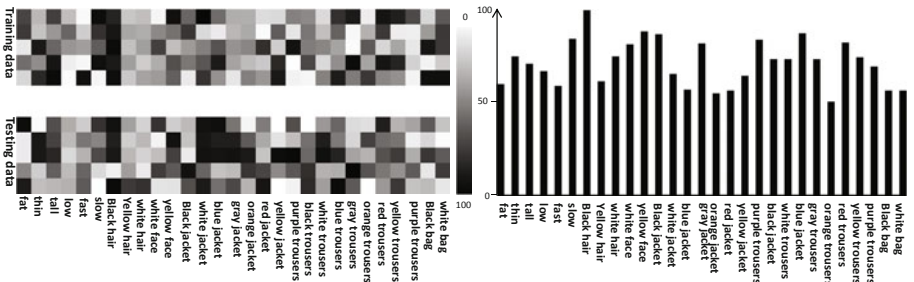
Fig. 2. (a) 5 training samples and 5 testing sample of human-attribute matrix in our dataset (b) the results of attributes learning

Feature Representation. To obtain the local descriptors, images are convolved with the filter bank proposed in[8], which is a combination of 3 Gaussians, 4 Laplacian of Gaussians, and 4 first order derivatives of Gaussians, and was shown to have good performance for object categorization. Instead of only computing visual words at interest points as in[7], we evenly sampled grid space at 10×10 pixels for a given image. Moreover, to maintain the importance of color in our attributes representation, unlike previous work, we computed the patch on three channels RGB and quantized into a visual word. The codebook, with the size W, is obtained by k-means.

3.2   Implementation Details and Results

Part 1: Performance of Body and Appearance Attribute Learning. Fig 2(a) show the real attributes value of 5 training models and 5 testing samples. The responses of 10 test persons were averaged to determine the real-valued association strength between attributes and human. The darker the boxes, the higher is the attribute correlated to the human. Binary attributes are obtained by thresholding at the overall matrix mean. The attribute #1, #3 and #5 are the antonym of attribute #2, #4 and #6 and nobody would consider a person fat and thin at the same time, thus the sum of the probability of the attributes equals to 100%. Attribute #7 *black hair* is almost 100% for every person except for test sample #4. *Yellow face* and *white face* is hardly to distinguish, thus the real value is more complicated than the body attributes. Different appearance attributes which describe the jacket and trousers are allowed to exist at the same time, because not the total clothes have only one color.

Fig.2(b) shows the learning results of the attributes. *Black hair* achieves the highest accuracy of nearly 100%, because such attribute appears in almost all the images and around the same position. The attributes *black bag* and *white bag* achieves the lowest accuracy, because the bag hardly appears in the training data, and the position changed a lot both on different views and different human.

Part 2: Overall Performance of Human Identification. In Sect 2.3, we introduced *Predict-Taken* and *Predict-Not-Taken* on attribute-loss, which allow the model to predict unseen attributes on different orientations. In the following, we evaluate both methods by applying them to our pedestrian dataset. For *Predict-Taken* method, we apply the average value of all learned attribute as the predict results of the lost ones. For *Predict-Not-Taken* method, we apply the minimal value of all learned attribute as the predict result of the lost one. Then, we train a non-linear SVM to predict the voter-candidate value as introduced in Sect 2. The weight parameter is determined by the frequency of the attribute normalized by all attributes without spatial limits in the training data. Using the 40 short videos, the sample pairs of different orientations from one person are selected as the positive samples, and different person without considering the orientations are selected as the negative samples of the SVM. In order to obtain probability estimates, we perform the SVM training using only 90% of the training samples and use the remaining training data as validation set for *Platt scaling*[3,9].

For testing, we select 3 clips for each one of the five pedestrian. Thus, each pedestrian has about 3 positive test pairs (employ permutation)and 18 negative test pairs for each human. As shown in Table 1 this results in a multi-person accuracy of 76.2% for "Predict-Not-Taken" as measured by the mean of the true positive and true negative, a multi-person accuracy of 84.8% for "Predict-Taken", and a multi-person accuracy of 89.5% for mixture of the two schemes. Clearly, the performance of the "Predict-Taken" is higher than "Predict-Not-Taken" of nearly 8%. On the other hand, the performance of mixed scheme is higher than the other two, higher performance on identification rate requires further attributes selection.

Table 1. Human identification results

Methods	True positive	False positive	True negative	False negative
Predict-Not-Taken	8	18	72	7
Predict-Taken	11	12	78	4
Mixed	13	9	81	2

4   Conclusion

In this paper, we propose a new identification method to construct an attribute model for human identification in a multi-camera environment. The proposed model is robust to orientation and environment. The main contributions of our work are summarized as follows. (1) An attribute based method for human identification. (2) A probabilistic model with spatial and view information to learn appearance attributes. (3) Two schemes on attributes-loss. In addition, the experimental results show that the proposed method is robust for surveillance

References

1. Liu, Z., Malave, L., Sarkar, S.: Studies on silhouette quality and gait recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2004)
2. Chellappa, R., Roy-Chowdhury, A.K., Kale, A.: Human identification using gait and face. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007)
3. Lampert, C.H., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2009)
4. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2009)

5. Yao, J., Odobez, J.M.: Multi-camera 3d person tracking with particle filter in a surveillance environment. In: European Signal Processing Conference, pp. 363–388 (2008)
6. Yao, J., Odobez, J.M.: Multi-layer background subtraction based on color and texture. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007)
7. Cao, L., Fei-Fei, L.: Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007)
8. Wang, X., Grimson, E.: Spatial latent dirichlet allocation. In: Conference on Neural Information Processing Systems, vol. 20 (2007)
9. Platt, J.C.: Probabilities for SV machines. In: Conference on Neural Information Processing Systems, pp. 61–74. MIT (1999)

Visual Information of Endpoint Position Is Not Required for Prism Adaptation of Shooting Task

Takumi Ishikawa and Yutaka Sakaguchi

University of Electro-Communications, Graduate School of Information Systems,
Laboratory for Human Informatics,
1-5-1, Chofugaoka, Chofu, Tokyo 182-8585, Japan
t.ishikawa@hi.is.uec.ac.jp, sakaguchi@is.uec.ac.jp
http://www.hi.is.uec.ac.jp

Abstract. Humans can easily adapt to a visually distorted environment: We can make correct movements after a few dozens of actions with visual guidance in the new environment. However, it is unclear what visual information our brain uses for this visuo-motor adaptation. To answer this question, we conducted a behavioral experiment of prism adaption of a ball shooting task, with manipulating visual information of the ball. We found that prism adaptation occurred when the position of ball impact (or endpoint) was not visually presented. A similar result was replicated in a modified experimental setup where the vision of the body was completely eliminated. These results imply that the error information at the time of hit/impact (i.e., the displacement between the target and the hit position) is not required for prism adaptation. This suggests that the visual information of on-the-fly ball trajectory can be utilized for prism adaptation.

Keywords: visuo-motor transformation, prism adaptation, motor learning, virtual shooting task, feedback delay.

1 Introduction

Our brain transforms visual information into motor information in making an action to a visually given target, for example, in reaching/grasping an object and shooting a target. Considering that human can modify the movement according to the change in the visual environment, our brain presumably updates the visuo-motor transformation adaptively. The learning mechanism of this transformation is an important topic in brain science, and a number of experiments have been reported [1,2,3,4], where visual environment was distorted by the wedge prism and VR devices. In concrete, a subject performs reaching [2] and shooting [3] with wearing goggles with wedge prisms. Although the endpoint of reaching/shooting is displaced by the amount of the visual shift just after wearing the prism goggles, the displacement gradually decreases and the subject correctly reaches/shoots the target after a few dozens of trials.

It is commonly accepted that prism adaptation is driven by some error signal contained in visual information. However, it is unclear what the essential source

of error signal is for the adaptation. In reaching task, for example, the displacement between the target and endpoint (i.e., task error) provides a kind of error signal. However, if our brain modifies the relationship between the motor command and the resultant endpoint (i.e., forward model), the error signal should be given by the difference between the predicted endpoint (based on the forward model) and the actual endpoint, not by the difference between the target and endpoint. The finding that the prism adaptation occurred when the target was displayed only instantly (and not displayed at the task end) (unpublished data) suggested that the "real" visual displacement between the endpoint and target is not required for adaptation. Anyhow, we know little about what kind of error information our brain actually uses for the prism adaptation.

In line with this discussion, the present study deals with the question whether the visual information of the endpoint is required for prism adaptation. To answer this question, we ran behavioral experiments of a ball shooting task. A subject wore shutter-controlled goggles which dynamically manipulated the visual information during the task. In the first experiment, we deprived of visual information from the midway to the end of the ball trajectory (that is, the subject could not see the impact position), and compared the magnitude of adaptation between the deprived-vision condition and full-vision condition. In the second experiment, we conducted a similar experiment where the visual information of the subject arm/hand during shooting action was completely eliminated. The results of these experiments showed no difference in magnitude of adaptation between two conditions, suggesting that the visual information of the impact position was not required for prism adaption of ball shooting task. We will discuss the implication of the present result.
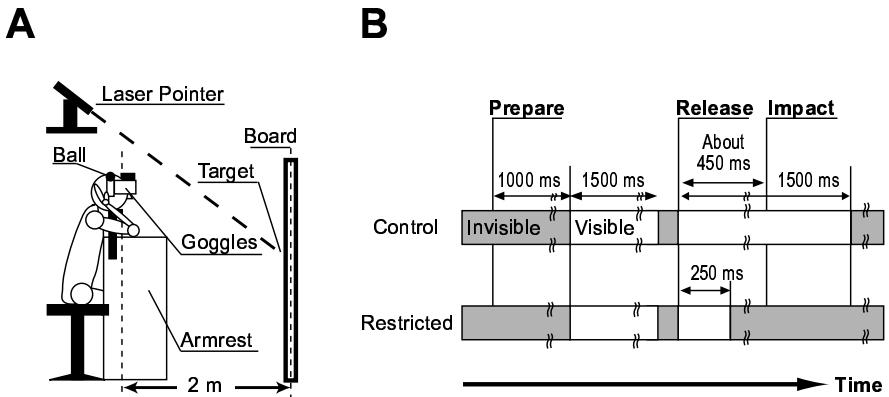
Fig. 1. Experimental Setup in Experiment 1

2   Experiment 1

2.1   Apparatus

A subject seated on a chair with his/her head restrained by a chin rest (Fig. 1 A). A subject wore goggles with ferroelectric liquid crystal (FLC) shutters (FE-1, Cambridge Research Systems) on the head, which controlled his/her sight. The response time of the FLC shutter was 0.1 ms and its transmission rate was 30% in the open condition, and its contrast ratio between open and close conditions was 1000:1. A pair of removable wedge prisms (about 9 diopter) were attached to the goggles. Subjects wore a special grove for detecting a ball in the hand. Rubber balls (4 cm in diameter, 50 g) were wrapped by aluminum foil, and the grove detected the existence of a ball using electric current. The subject task was to throw out a ball to a target by overhand. The target was specified by a green laser spot projected to the fiberboard (hereafter to be called the board) which stood 2 m in front of the subject. The position of the spot was controlled by the experimental program. A shock sensor detected the impact of the ball hitting.

2.2   Method

Three subjects participated in the experiment.

Figure 1 B shows the diagram of an experimental trial. The FLC shutter was closed at the beginning of a trial. A subject picked up a ball from the ball holder (placed near the subject's temple) with the thumb and forefinger. One second after the grove detected the ball, the shutter was opened and the target became visible. The shutter was closed again in 1.5 second. The subject started the shooting when the shutter was closed. The shutter was opened again when the ball was released from the hand. The shutter was closed at a time specific to the experimental condition. The next trial starts 5 seconds after the ball reached the board. One experimental block consisted of 35 trials. First 5 trials were "*pre-adaptation* period" which gave the baseline of the performance. Next 15 trials were the "*adaptation* period" where the wedge prisms were attached to the subject's goggle, and the subjects performed the shooting task in a visually shifted environment. The last 15 trials were "*de-adaptation* period", where the wedge prisms were detached.

2.3   Conditions

We prepared the two conditions. In *control* condition, the shutter closed 1500 ms after the ball was released. In this condition, subject could see the hit location because the ball reached the target board in about 450 ms after the ball release. In *restricted* condition, on the other hand, the shutter closed 300 ms after the ball release so that the subject could not see the hit location.

Each subject performed two sessions of three blocks. Orders of the two experimental conditions are counter-balanced among the subjects.

2.4   Result and Discussion

Figure 2 shows the result of experiment, where horizontal errors relative to the target are plotted against trial sequence.

The line shows the mean of 9 sessions (3 subjects x 3 blocks). The arrow indicates the amount of *aftereffect* (i.e., mean of the error of the first trial in the *de-adaptation* block).

We calculated the amount of *aftereffect* by the horizontal error in the first trial of the *de-adaptation* period, and compared it between *control* and *restricted* conditions. As shown in Fig. 3, they showed no significant difference ($p = 0.9502$,
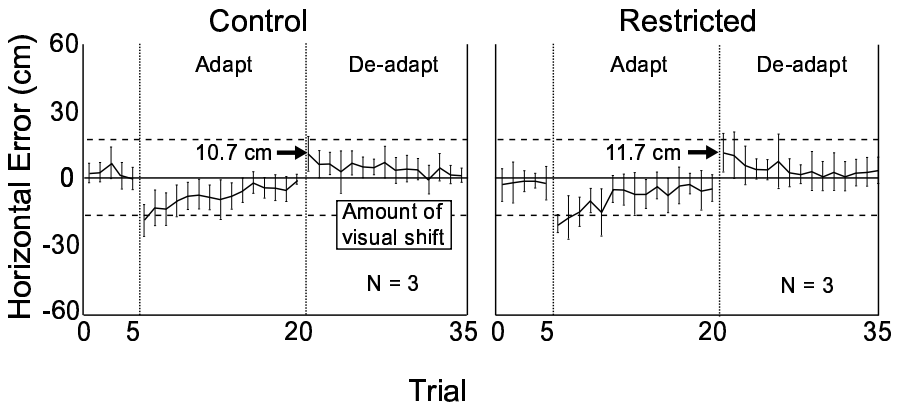
Fig. 2. Learning Curves in Experiment 1
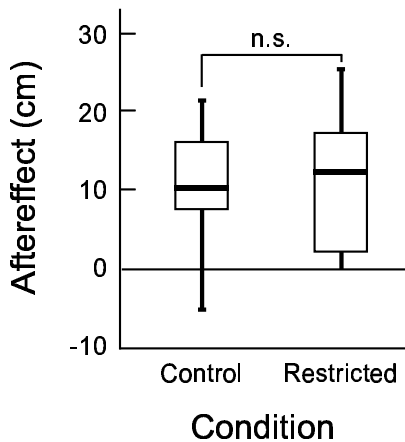
Fig. 3. Amount of Aftereffect in the Experiment 1

Wilcoxon rank-sum test). Of course, the amount of aftereffect (the error in the first trial of *de-adaptation* period) was significantly greater than zero in both conditions ($p < 0.01$).

This result shows that prism adaptation occurred irrespective of the experimental conditions (i.e., durations of visual information).

The present result indicates that prism adaptation was not degraded even if the real visual information on the shooting error (i.e., the displacement between the target and hit position) was not provided. Then, a question can be raised what visual information our brain used for the adaptation. One possibility is that the brain used visual information of on-the-fly ball trajectory. There are two possible ways for using this visual information for adaptation. First, the brain guesses the hit position from the observed ball trajectory and utilizes the displacement between the predicted hit position and the observed target [5]. Second, the brain predicts the ball trajectory and obtains the difference between the predicted trajectory and actual trajectory. We do not discuss which of these ways is true. Another possibility is that our brain might utilize the visual information of the body movement for adaptation. It is plausible that the posture of the arm/hand in throwing action could be correlated with the ball trajectory. Thus, it is possible that the visual information of the arm movement might give a clue for the prism adaptation. In order to examine the second possibility, we ran the second experiment explained below.
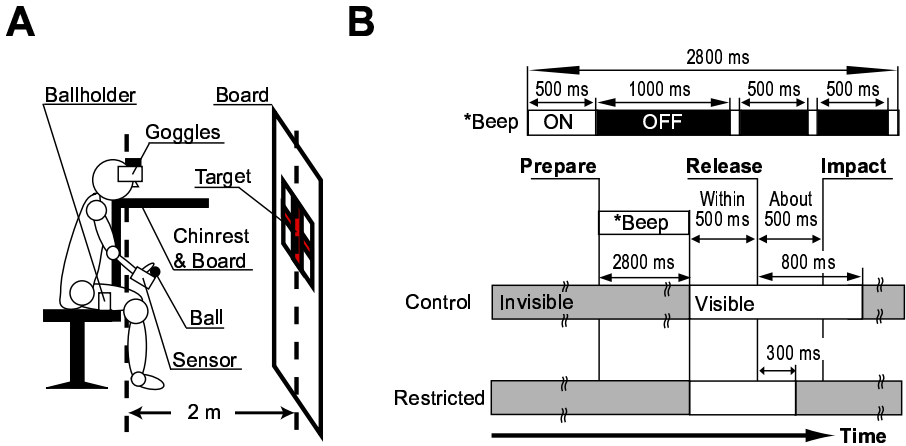
Fig. 4. Experimental Setup in Experiment 2

3   Experiment 2

3.1   Method

Four subjects participated in the experiment.

Experimental apparatus (Fig. 4 A) was almost the same as in the first experiment, but different in several points, including the magnitude of prism shift and throwing manner (overhand vs. underhand). Shutter goggles were the same as the first experiment. Shutter goggles were fixed to a head-chin rest. Removable wedge prisms (about 17 diopter) were attached to the goggles. A horizontal plate was placed near the subject's chin which prevented the subject from seeing his/her arm and hand. The subject wore a special finger grove for detecting a ball held in the hand (its detail was different from that used in the first experiment). Rubber balls (4 cm in diameter, 50 g) were wrapped by aluminum foil. The subject picked up a rubber ball from a ball holder (located around the thigh), and threw the ball in an underhand manner. The shock sensor installed on the board for detecting the ball impact. A microcomputer gathered the sensor data and controlled the actuator devices (including sound beeps) in a real-time manner. In this experiment, the target was given by a colored paper (4 cm x 4 cm) attached on the board. The position of the target was displaced between the without-prism trials and with-prism trials so that visual target position was maintained at the same position irrespective of the prism condition.
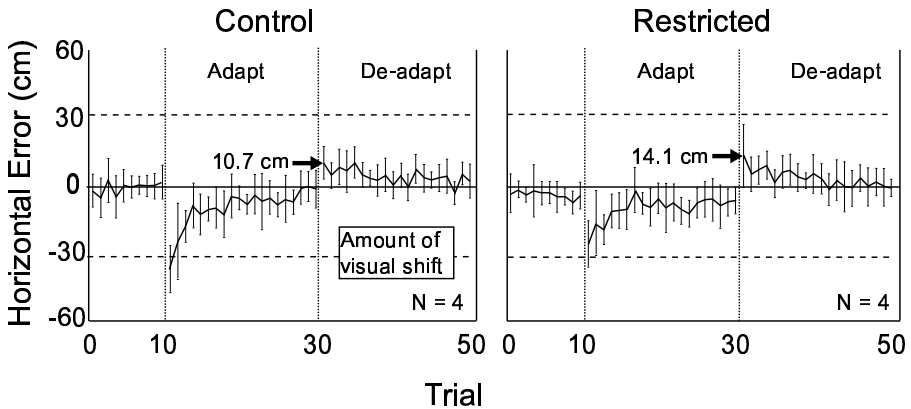
Fig. 5. Learning Curves in Experiment 2

Figure 4 B shows the diagram of the visual information presentation within an experimental trial. One second after the subject held a ball with the thumb and the implement, three beeps rang with intervals of 600 ms, and the FLC shutter opened at the offset of the third beep. The subject was asked to throw the ball within 500 ms after the third beep. The shutter was closed at a time specific to the experimental condition. The next trial started after 3 seconds after the ball reached the board. Each block consists of 50 trials as in the first experiment. Concretely, "*pre-adaptation* period" were 10 trials, "*adaptation* period" were 20 trials, and "*de-adaptation* period" were 20 trials.
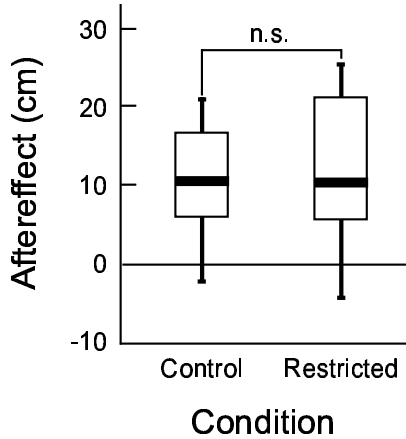
Fig. 6. Amount of Aftereffect in the Experiment 2

3.2   Conditions

We prepared the *control* condition and *restricted* condition as in the first experiment. In the *control* condition, the shutter closed 800 ms after the ball release, whilst it closed 300 ms after the ball release in the *restricted* condition. Each subject performed four blocks (2 conditions x 2 shift-directions). Orders of the experimental conditions were counter-balanced.

3.3   Result and Discussion

Figure 5 shows the learning curves in the two conditions. The amount of *aftereffect* showed no significant difference between *control* and *restricted* conditions ((Fig. 6, $p = 0.8785$). The amount of *aftereffect* was significantly larger than zero in both conditions ($p < 0.01$). Again, the prism adaptation occurred irrespective of the vision conditions. Thus, real visual information on the task error was not required for the prism adaptation of shooting task. In addition, the brain does not rely on the visual information on the body movement. This in turn implies that visual information of the on-the-fly trajectory is sufficient for the prism adaptation.

4   Concluding Remark

The present results confirmed that the prism adaptation of the ball shooting task occurs if no visual information is provided on the hit location (or shooting error). This suggests that for prism adaptation of a shooting task, our brain may use visual information of the flight trajectory. Indovina, et al. [6] suggested some predictive mechanisms of physical laws of motion may be equipped in the

human brain. In a shooting task, what information does the brain utilize for the adaptation? As mentioned above, we hypothesized two possibilities. First, our brain may estimate the hit position based on the on-the-fly ball trajectory (using the forward model of ball motion), and compare the estimated hit position and the observed target, giving the endpoint error (or task error). Second, our brain may compare the predicted ball trajectory (based on the forward model of ball trajectory brought by a shooting action) and observed ball trajectory and use their difference for adaptation. It is open for further investigation which hypothesis is true, but the answer to this question is indispensable for building a computational model for visuo-motor adaptation.

References

1. Stratton, G.M.: Some preliminary experiments on vision without inversion of the retinal image. Psychological Review 3, 611–617 (1896)
2. Kitazawa, S., Kohno, T., Uka, T.: Effects of delayed visual information on the rate and amount of prism adaptation in the human. J. Neuroscience 15, 7644–7652 (1995)
3. Martin, T.A., Keating, J.G., Goodkin, H.P., Bastian, A.J., Thach, W.T.: Throwing while looking through prisms I. Focal olivocerebellar lesions impair adaptation. Brain 119, 1183–1198 (1996)
4. Newport, R., Jackson, S.R.: Posterior parietal cortex and the dissociable components of prism adaptation. Neuropsychologia 44, 2757–2765 (2006)
5. Pelisson, D., Prablanc, C., Goodale, M.A., Jeannerod, M.: Visual control of reaching movements without vision of the limb. II. Evidence of fast unconscious processes correcting the trajectory of the hand to the final position of a double-step stimulus. Exp. Brain Res. 62, 303–311 (1986)
6. Indovina, I., Maffei, V., Bosco, G., Zago, M., Macaluso, E., Lacquaniti, F.: Representation of visual gravitational motion in the human vestibular cortex. Science 308, 416–419 (2005)

Q-Learning with Double Progressive Widening: Application to Robotics

Nataliya Sokolovska, Olivier Teytaud, and Mario Milone

INRIA Saclay, CNRS UMR 8623 & LRI, Université Paris Sud, Orsay, France

Abstract. Discretization of state and action spaces is a critical issue in Q-Learning. In our contribution, we propose a real-time adaptation of the discretization by the progressive widening technique which has been already used in bandit-based methods. Results are consistently converging to the optimum of the problem, without changing the parametrization for each new problem.

Keywords: Q-Learning, discretization, applications.

1 Introduction

In a large number of real world applications it is intractable to estimate a model. Q-Learning is a well known model-free reinforcement learning algorithm, where Q-values – which estimate the expected reward for taking a particular action in a given state – are learnt. However, in the approach it is assumed that the domain is discrete, or discretized. If the state and/or action spaces are continuous, the application of the Q-Learning is not straightforward. If the state/action domains are continuous (or very large), it becomes hardly possible to keep (and to update) a look-up table which contains Q-values for each state-action pair. Besides discretization approaches [1,2,3], including adaptive techniques, there exist a number of techniques applied to the reinforcement learning which allow to work with continuous values. To discretize a continuous state and action space is a challenge, since if a discretization is too rough, it will be impossible to find the optimal policy; if a grid is too fine, the generalization will be lost.

Among the state-of-the art approaches are the following discretizing (and often feature and model selecting) approaches. A historical but still actively exploited approach is CMAC (Cerebellar Model Arithmetic Computer) that has been introduced for robotics [4,5]. In CMAC Q-Learning the state space is partitioned into tiles, which are binary features. A parameter (or weight) θ is associated with each tile, and Q-values are not kept in a look-up table but are represented by a parametric family of functions, parametrized by the vector Θ. CMAC Q-Learning is similar to a neural network. Overall, the introduction of neural networks into Q-Learning to process continuous states has been actively studied, see e.g., [6]. If the number of tiles is quite large, the computational complexity of such parametric approaches can be high.

Vector Quantization Q-Learning (VQQL) [7,8] produces a compact representation of the state domain via clustering of simulated states. The drawback of the method is that a refinement of the grid is not foreseen.

Recently in [9] the Two Steps Reinforcement Learning (2SRL) has been introduced, where decision (and regression) trees are used to perform the state space discretization. The algorithm is based on two alternating steps; in the first phase some discretization of the state domain is produced, in the second one a current policy is improved. The reported drawback of the method is the requirement for the discrete reward function.

The important problem is to refine the discretization grid of states (and actions) adaptively, especially around the areas of interest, e.g., around the goal. If the initial grid is rather coarse, and if all vertices of the grid are far enough from the goal, it is possible that an agent never reaches a goal. An adaptive approach to refine the initial grid has been recently proposed by [10]. The idea is to provide pseudo-goals which lie on the vertices of the initial grid. It has been shown that the method is efficient, however, its serious disadvantage is that the knowledge of a location of a goal is required. The initialization of the grid with the pseudo-goals, which are in a proximity to the true goal, is not obvious.

In this contribution, we propose a technique, inspired by methodologies developed in Monte-Carlo Tree Search, for directly working in the continuous setting. To the best of our knowledge, this is the first dynamic discretization approach handling both continuous states and continuous actions - which is critical for many important applications.

The paper is organized as follows. Section 2 presents progressive widening Q-Learning. Section 3 provides the results of our experiments on both a synthetic reinforcement learning task and on a realistic problem of a robot navigation is a 3-dimensional partially observable environment. Concluding remarks and perspectives close the paper.

2   Progressive Widening Q-Learning

In this section, we describe briefly the Q-Learning approach. We discuss its limits with respect to continuous problems, and introduce progressive widening Q-Learning procedure.

2.1 Q-Learning Approach

To solve a goal-planning task means to find an optimal policy, i.e. a policy $\pi^\star$ that is equal or better (in terms of cumulated expected reward) than any other policy π. It is known [11] that optimal policies share the same optimal action-value function $Q^\star$. Given a set of states $\mathcal{S}$ and a set of actions $\mathcal{A}$, optimal action-value function is defined as

$$Q^{\star}(s,a) = \max_{\pi} Q^{\pi}(s,a)$$

$$= \mathrm{E}\{r_{t+1} + \gamma \max_{a'} Q^{\star}(s_{t+1}, a')|s_t = s, a_t = a\}$$

$$= \sum_{s'} \mathcal{P}_{ss'}^a \left(\mathcal{R}_{ss'}^a + \gamma \max_{a'} Q^{\star}(s', a') \right), \tag{1}$$

where $\mathcal{P}_{ss'}^a = p(s'|s,a)$, $\mathcal{R}$ is the reward, $s, s' \in \mathcal{S}$, and $a, a' \in \mathcal{A}$. In other words, the action-value function $Q : \mathcal{S} \times A \longrightarrow \mathbb{R}$ defines the quality of each (state, action) pair.

Q-Learning is a general term for approaches which compute the expected reward given an action a in a given state s, and allow to choose an action maximizing the reward value. The strength of Q-Learning methods is that they do not require any knowledge of a model of environment $\mathcal{P}_{ss'}^a$, which is not available in a number of real-world applications.

An example of a policy is the greedy policy, given by

$$\pi(s) = \arg\max_{a} Q^{\pi}(s,a), \tag{2}$$

which we use in the following.

One-step Q-Learning is proposed by [12]. The approach is based on the following update rule:

$$Q(s,a) = Q(s,a) + \alpha \left(r + \gamma \max_{a'} Q(s', a') - Q(s,a) \right), \tag{3}$$

where α is usually called learning rate, $\alpha \in]0;1]$, and γ – discount factor, $\gamma \in [0;1[$. The complete reinforcement learning procedure is drafted as Algorithm 1.

Algorithm 1. One-step Q-Learning

Initialize $Q(s,a)$
for each episode **do**
   Initialize s
   for each step of episode **do**
     Choose a (ϵ-greedy policy derived from Q)
     Take action a, observe r and s'
     $Q(s,a) = Q(s,a) + \alpha \left(r + \gamma \max_{a'} Q(s', a') - Q(s,a) \right)$
     $s = s'$
   end for
end for

2.2 Applying Double Progressive Widening to Q-Learning

Since we can not enumerate all possible states and all possible actions, we exploit the idea to use a table of Q values which is not static. We explore and add states

and actions to the Q-table progressively. Under double progressive widening we mean that we increase the number of explored states and actions.

We can naturally apply the double progressive widening procedure to the Q-Learning framework. Since our state and action spaces both are continuous, some discretization has to be done, if we want to apply the Q-Learning directly. At the same time, both the state and action spaces should be well explored to achieve some reasonable cumulated reward. For this, we slowly increase the number of states and actions, by the progressive widening technique successfully used in bandit-based algorithms [13,14,15]. When a state is sufficiently highly visited, compared to the number of times the previous action has been tried in the previous state, then it is added in the discretization of states; and when the number of visits of a state is sufficiently large, compared to the pool of actions already considered, then a new action is added. The discretization of states and actions is carried out based on the Euclidean distance. A newly observed state (action) gets the same discrete value as its closest state (action) in the already explored set of states $\mathcal{S}$ (set of actions $\mathcal{A}$).

The approach we use is drafted as Algorithm 2; λ is the progressive widening parameter associated with states exploration, and λ' – with exploration of decisions. Both parameters are equal in our experiments, so that we do not introduce a bias by a highly tuned parameterization; interestingly, we will see that some values of $\lambda = \lambda'$ are good for all our tests.

Algorithm 2. Progressive Widening Q-Learning

Initialize $\mathcal{S}$ – set of explored states
Initialize $\mathcal{A}$ – set of explored actions, specific for states
Initialize $Q(s, a)$ – Q-values
Initialize parameters $C > 0$, $\lambda \in]0, 1[$, $\alpha \in]0; 1]$, $\gamma \in [0; 1[$
for each episode **do**
   Initialize s
   for each step of episode **do**
      $nbVisits(s) = nbVisits(s) + 1$
      $k = \lceil CnbVisits(s)^{\lambda} \rceil$
      Choose action a from $\{a_1, \ldots, a_k\}$ associated with s using Eq. (2)
      Update the number of visits $nbVisits(s, a) = nbVisits(s, a) + 1$
      Take action a, observe s' and r
      if $\left(\lceil CnbVisits(s, a)^{\lambda'} \rceil > \#\mathcal{S} \right) \& \left(s' \notin \mathcal{S} \right)$ **then**
         $\mathcal{S} = \mathcal{S} \cup s'$
      end if
      Update $Q(s, a)$, using Eq. (3)
      $s = \arg\min_{s'' \in \mathcal{S}} \|s'' - s'\|$
   end for
end for

---

3   Experiments

In this section, we illustrate efficiency of the proposed progressive widening *Q*-Learning on a synthetic problem called Treasure Hunt, and on a realistic – robot navigation – task.

3.1   Treasure Hunt Problem

The problem considered in this section is an artificial problem which allows us to demonstrate clearly that the double progressive widening *Q*-Learning is a powerful approach. Treasure Hunt is a two-dimensional problem, i.e., an agent moves in a two-dimensional environment, with the dimensions 15×15. The aim of the agent is to discover the treasure. Both the agent and the treasure are always initialized in the same coordinates (the agent is initialized in the lower left corner of the 15×15 room, and the treasure is located in the upper right corner). The agent knows his position at each time step. The reward equals 1000 when the treasure is reached, otherwise the instantaneous reward equals -1.

To consider different scenarios, we make use of three variations of the Treasure Hunt framework:

1. Treasure Hunt as described above.
2. An obstacle is added, i.e. a hole is added in the center of the two-dimensional space. If the agent falls into the hole, the reward is -500.
3. Uniform noise (on states) is added.

The difficulty is the fact that both states (positions of the agent) and actions (or decisions of the agent) are continuous. In our experiments, we compare the standard *Q*-Learning without progressive widening with the method proposed above. To perform experiments with the standard *Q*-Learning we discretize the state and action spaces using some grid of a constant size. The progressive widening parameter λ controls how many and how fast new states and actions are added to the table, which contains states $\times$ actions *Q*-values. In our experiments, we apply the same progressive widening parameter value to states and actions, i.e., $\lambda = \lambda'$. Note, that when $\lambda = 0$, neither new states, nor new actions are added, and therefore, the results are equivalent to ones of the standard *Q*-Learning. We can start the progressive widening procedure from empty sets for states and actions. In the experiments, we have considered two cases: we start the learning procedures (*Q*-Learning and progressive widening *Q*-Learning) from 1 pre-simulated state and action; and from 5 pre-simulated states and actions. We use the following *Q*-Learning parameters is our experiments: $\alpha = 0.15$, $\gamma = 0.85$.

The following figures illustrate our results – the mean of the cumulated reward – on different scenarios of the Treasure Hunt problem and for difference progressive widening (PW in the legend) parameter values. The number of Monte-Carol simulations on the plots is 500. Figure 1 is the basic case, where there are no any obstacles and the transitions are completely deterministic (no noise). Figure 2 demonstrates the case with the noise. Figure 3 is the variation with the hole in the center of the 2-dimensional search space, and Figure 4 is the

scenario with the obstacle and with the noise. We can conclude that the standard Q-Learning with 1 pre-simulated state and action copes quite bad with the task. The Q-Learning which disposes 5 states and 5 actions ameliorates the performance when the number of the Learning episodes increases. It is obvious that the double progressive widening Q-Learning is much more efficient. Note, that the progressive widening parameter plays a significant role, $\lambda = 0.5$ allows to add more states and actions than $\lambda = 0.25$, and hence makes the discretization of the state and action spaces more adapted. Overall, the problem is rather simple and can be discretized in a small number of states and actions.
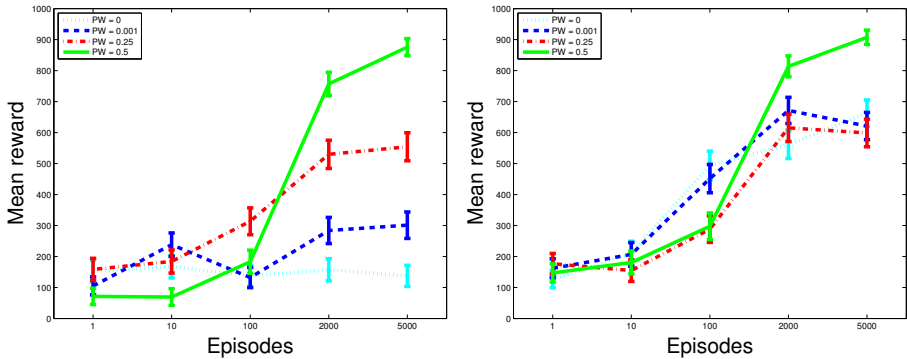
Fig. 1. Treasure Hunt task. Without noise and without obstacles; left: starting with 1 state and 1 action; right: starting with 5 states and 5 actions. Note that while applying the parameter 0, the double progressive widening boils down to a non-adaptive discretization.

3.2   Robot Navigation in a 3D Partially Observable Environment

Autonomous robot navigation is a challenging task, especially in an environment which is partially observable. The values of the Q-Learning parameters and the number of simulations are the same as in the previous section.

The 3D simulator we use in our experiments has been developed at IDIAP[1]. The simulator models the 3D environment which resembles one designed for computer video games. An agent is placed in a virtual room, and the goal is to teach it to touch the red flag. In every training (and testing) episode, the robot and the flag are placed randomly. The described task is a typical problem which can be solved by reinforcement learning. If the robot touches the walls, the instantaneous reward equals -1, if the red flag is reached the reward is $+10$, otherwise the reward is 0 at each time step.

The difficulty of the task is that the environment is not fully observable, it is partially observable. The robot does not know its position. The agent has to

---

[1] http://www.idiap.ch/
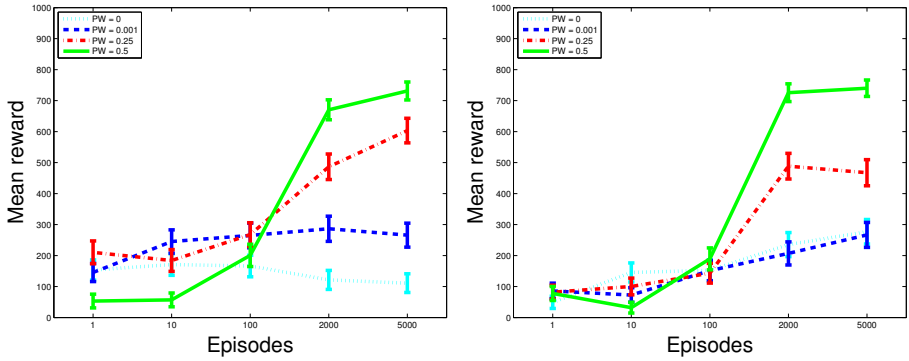
Fig. 2. Treasure Hunt task. With noise and without obstacles; left: starting with 1 state and 1 action; right: starting with 5 states and 5 actions. Note that while applying the parameter 0, the double progressive widening boils down to a non-adaptive discretization.
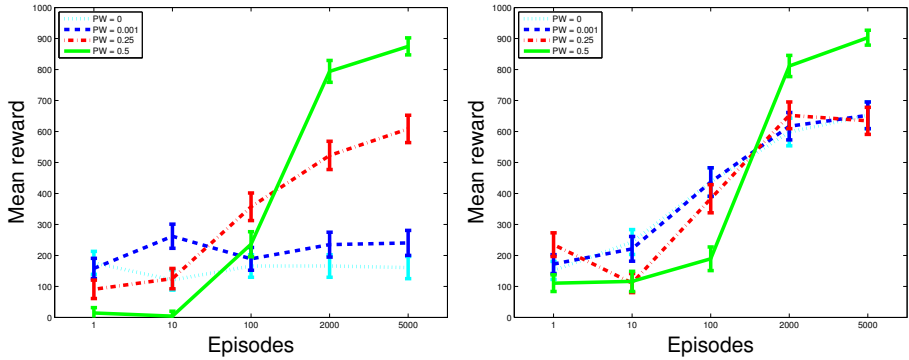
Fig. 3. Treasure Hunt task. Without noise and with obstacles; left: starting with 1 state and 1 action; right: starting with 5 states and 5 actions. Note that while applying the parameter 0, the double progressive widening boils down to a non-adaptive discretization.

deduce where it is and what it has to do (i.e., which action should be taken) based on video images it gets on each time step. Figure 5 illustrates two typical observations of an agent (on the left: the robot does not observe the goal, on the right: the robot sees the flag).

On each time step, the agent gets a video image. Using the a priori knowledge that the flag, the robot is looking for, is red, we apply the image processing technique to extract the information, whether the agent sees the flag, whether the flag is observed on the right/left/in front of the robot. We have introduced an additional reward for the training phase only. If the flag is observed on the right or on the left, the supplementary reward is +2.5, if the robot can observe the goal just in front of it, the supplementary reward equals +5. Figure 6 shows the dependence of the cumulated reward (test phase) on the learning time.
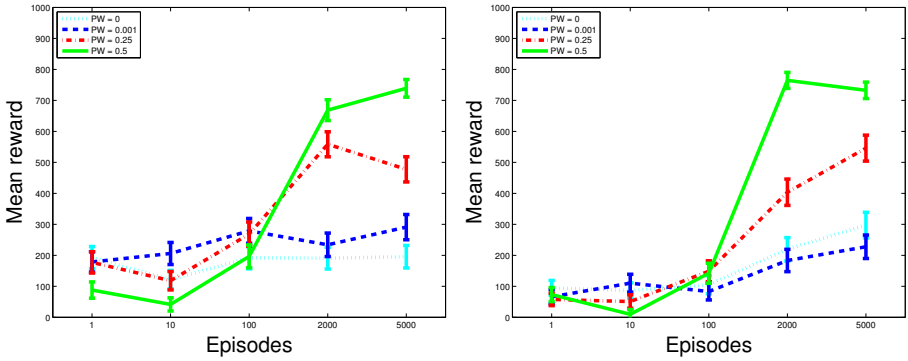
Fig. 4. Treasure Hunt task. With noise and with obstacles; left: starting with 1 state and 1 action; right: starting with 5 states and 5 actions. Note that while applying the parameter 0, the double progressive widening boils down to a non-adaptive discretization.
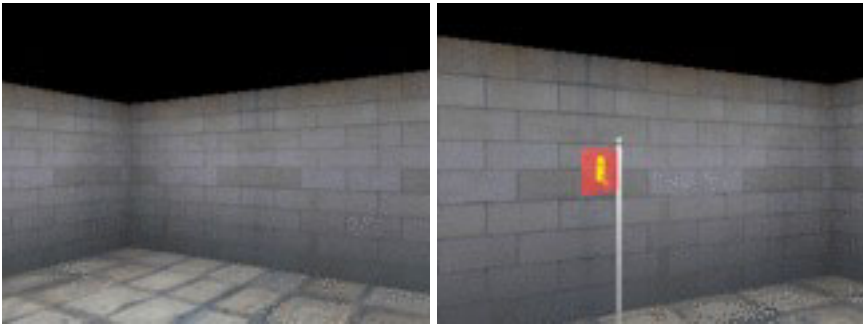
Fig. 5. 3D environment designed by the simulator. On the left: the robot does not observe the goal, on the right: the robot sees the flag.
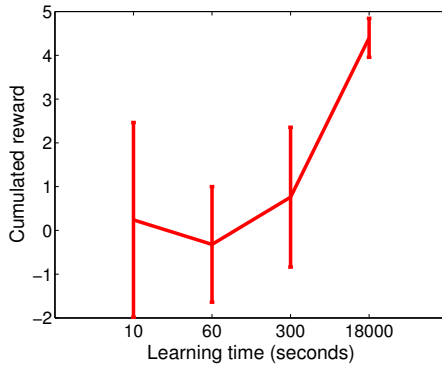
Fig. 6. Navigation of the robot in a 3D environment: cumulated reward (testing phase); $\lambda = 0.5$

In the Appendix we provide video images of a typical exploitation trajectory of the agent. The robot and the goal are initialized and placed randomly, and the agent is approaching the the red flag.

4   Conclusion

We proposed, to the best of our knowledge, the first approach for handling, without fixed discretization, both a continuous state space and a continuous action space. While we have no consistency proof, we believe that results as in [16] can be used for proving the consistency of our approach. Experimental results suggest that the algorithm efficiently adapts the discretization where it is needed, and that the "widening" principle, consisting in extending an edge when it is simulated more than n^λ times where n is the number of visits to the parent situation, is a stable methodology with coefficient λ around $\frac{1}{3}$ (interestingly, nearly the same constant as in [17] and [16] in different contexts).

References

1. Davies, S.: Multidimensional Triangulation and Interpolation for Reinforcement Learning. In: Advances in Neural Information Processing Systems (1997)
2. Munos, R., Moore, A.: Variable Resolution Discretization in Optimal Control. Technical report, Robotics Institute, CMU (1999)
3. Munos, R., Moore, A.W.: Variable Resolution Discretization for High-accuracy Solutions of Optimal Control Problems. In: IJCAI, pp. 1348–1355 (1999)
4. Albus, J.S.: A New Approach to Manipulator Control: the Cerebellar Model Articulation Controller. Journal of Dynamic Systems, Measurement, and Control 97, 220–227 (1975)
5. Burgin, G.: Using Cerebellar Arithmetic Computers. AI Expert 7 (1992)
6. Gaskett, C., Wettergreen, D., Zelinsky, A.: *Q*-learning in Continuous State and Action Spaces. In: Foo, N.Y. (ed.) AI 1999. LNCS, vol. 1747, pp. 417–428. Springer, Heidelberg (1999)
7. Gersho, A., Gray, R.M.: Vector Quantization and Signal Compression. Kluwer Academic Publishers (1991)
8. Stone, P., Sutton, R.S., Kuhlmann, G.: Reinforcement Learning for Robocup-soccer Keepaway. Adaptive Behavior 3, 165–188 (2005)
9. Fernández, F., Borrajo, D.: Two Steps Reinforcement Learning. International Journal of Intelligent Systems 2, 213–245 (2008)
10. Lampton, A., Valasek, J.: Multiresolution State-Space Discretization Method for *Q*-Learning. In: American Control Conference (2009)
11. Sutton, R.S., Barto, A.G.: Reinforcement Learning: an Introduction. MIT Press (1998)
12. Watkings, C.J.C.H.: Learning from Delayed Rewards. PhD thesis, Cambridge University (1989)

13. Couëtoux, A., Hoock, J.B., Sokolovska, N., Teytaud, O., Bonnard, N.: Continuous Upper Confidence Trees. In: International Conference on Learning and Intelligent Optimization (2011)
14. Coulom, R.: Monte-Carlo Tree Search in Crazy Stone. In: Game Programming Workshop (2007)
15. Rolet, P., Sebag, M., Teytaud, O.: Boosting Active Learning to Optimality: a Tractable Monte-Carlo, Billiard-based Algorithm. In: European Conference on Machine Learning (2009)
16. Wang, Y., Audibert, J.Y., Munos, R.: Algorithms for Infinitely Many-armed Bandits. In: Advances in Neural Information Processing Systems (2008)
17. Coulom, R.: Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search. In: Ciancarini, P., van den Herik, H.J. (eds.) Proceedings of the 5th International Conference on Computers and Games, Turin, Italy (2006)

Appendix: Exploitation Trajectory of the Agent
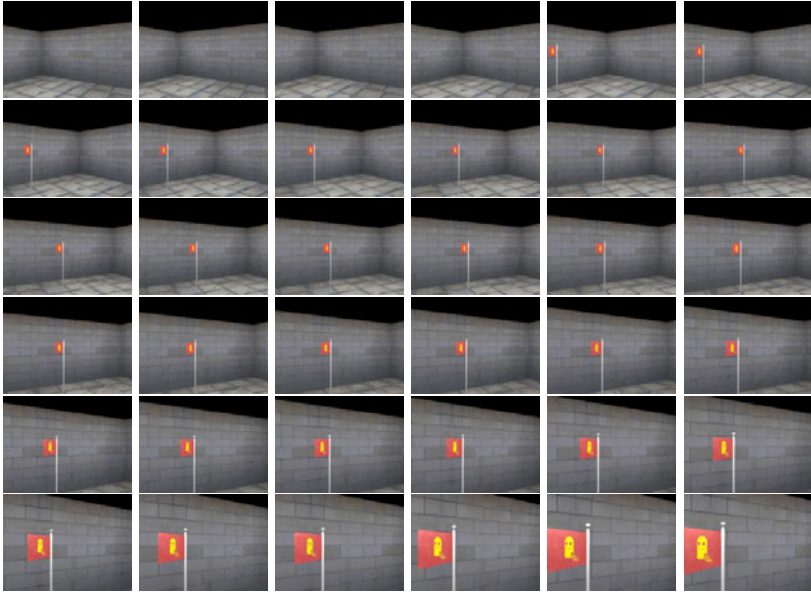
User Identification for Instant Messages

Yuxin Ding, Xuejun Meng, Guangren Chai, and Yan Tang

Harbin Institute of Technology Shenzhen Graduate School, Shenzhen 518055, China
yxding@hitsz.edu.cn, {xjmeng,grchai,ytang}@hotmail.com

Abstract. In this paper we study on recognizing user's identity based on instant messages. Considering the special characteristics of chatting text, we mainly focus on three problems, one is how to extract the features of chatting text, the second is how the user's model is affected by the size of training data, and the third is which classification model is fit for this problem. The chatting corpus used in this paper is collected from a Chinese IM tool and different feature selection methods and classification models are evaluate on it.

Keywords: Authorship, identification, classification, instant message.

1   Introduction

The goal of this paper is to determine whether an instant message is sent by the real sender indicated by the sender's ID. It belongs to the field of authorship verification. Authorship verification is used to determine whether an author (for whom we have of a corpus of writ-ing samples) is also the author of a given anonymous text. Usually it can be viewed as a multi-class, single-label text categorization from the point of the machine learning.

   With the wide application of instant messaging tools, instant messaging (IM) has become the most popular way of communication through internet. Unfortunately, this is also the reason why IM becomes a popular object to be attacked by hackers. Once an IM account is stolen by a hacker, the hacker can then send fraudulent messages to the victims' friends in the name of the victim. This brings the users of IM a serious security risk. Therefore, it is a meaningful work to verity the identity of an instant message sender. Currently as what we know, no similar research has been done in this field.

2   Related Work

2.1   Verification Problem

Some researchers look authorship verification as 'similarity detection' task, which determine whether they are generated by the same entity or not, without knowing the actual author. Pavelec[1] used this way to address the writer independent model which is possible to create a robust system even when few genuine samples per writer

are available. Following the same notion of verification, Van Halteren[2] has proposed a relatively different method called linguistic profiling.

There is another group of researchers who look the authorship verification as two-class text classification problem. For example, writer dependent or personal model adopted by [1] is based on one model per author. However, it is implemented using one-against-all strategy which means that positive examples and negative examples are all required to create this model. Luyckx and Daelem[3] also applied this kind "one vs. all" scheme to analyze the contribution of each kind of features.

Koppel & Schler[4] considered authorship verification as a one-class classification problem, and 'unmasking' method is proposed for authorship verification. However, this method needs long texts.

2.2   Features

Authorship analysis is based on the hypothesis: every author has the writing habit of his own, and this habit is defined in terms of different features. Usually two types of feature are employed, one is content-based features, and the other is stylometric features.

1. content-based features is relevant to a specific subject, if we want to determine the author of an anonymous text in a specific subject, this feature has been proved to be efficiently [5]. 2. stylometric features include three types of features: token-based feature, syntax feature and structural features.

Token-based features can be either word-based or character-based features. Word-based features include word length, sentence length [6], word frequency [7], word n-gram [8],   and vocabulary richness[9]. Character-based features include alphabetic characters count, uppercase and lowercase characters count ,digit characters count, letter frequencies, and punctuation marks count [9][10][11]. The most efficient feature of this measure is n-gram feature, and the application of this approach to authorship identification has been proven quite successful [7][12][13][14].

Syntax features, including function words [5][15][16][17], punctuations, and part of speech [18][19], can capture the writing style in sentence level, it is considered more reliable authorial features in comparison to the token-based features. Among them, function words are the best features to discriminate between authors.

Structural features which are applied to handle the layout and organization of a piece of writing have proved particularly important in analyzing online texts [9]. It includes the number of paragraphs or sentences, the average length of the paragraph, as well as some word structure. Similar features are introduced to the chat mining for gender prediction and authorship characterization problem [20][21].

Most of works mentioned above experimented with the literatures. However, the style of IM messages is very different from them. The real time nature of IM message produces unedited text; they are relatively casual compared with formal text. The most difficult problem is that IM messages are very short, and it is very hard to collect a large amount corpus for analyzing, in addition the real time nature of instant messaging requires to decide the identity of a user in a short time and using messages as small as possible. Therefore, we study on validating the identities of IM users.

3   User Identification for IM Messages

3.1   Feature Selection and Extraction

In this paper the chatting corpus we used is Chinese text. We choose the following features considering the special characteristic of Chinese chatting text:

Token-based features: tokens are defined as Chinese characters, punctuations, English words, and some other separated alphabets or symbols appeared in an instant message. In this paper we choose the following token-based features from a sample: the number of Chinese characters, the number of English words, the number of digits, frequency of punctuations, and the punctuation richness (how many types of punctuation are used in one sample).

In our experiments we found punctuation is effective in authorship verification for chatting text. When using IM to communicate with others, users are used to using characters they like to separate sentences. For example, some peoples like to use white space, while others may excessively use some special separators. Moreover, users always prefer to use what they like without considering whether the usage is correct in grammar. Useful punctuations are selected according to the frequency they occur in the text. The punctuation richness is calculated by the ratio of the number of punctuations to the total number of tokens in a sample.

Syntax features: Syntax features include frequency of function words and POS tag frequency. Firstly, we use Chinese Word Segmentation tool to find Chinese words, and then use Chinese POS tagger to assign POS tags to Chinese words. Function words are selected according to their POS tags. To select effective features of a user, we prepare two datasets for each user, one dataset contains only text messages of the user's own, and the other dataset contains messages randomly selected from chatting groups which include 2000 peoples. Information gain method is used to select effective function words that can represent the user's characteristics. We calculate the information gain of each function word and sort the function words in the user's own dataset in the decreasing order of their information gain, and select the top N function words as feature words.

POS tag features are selected according to their frequency in the user's own dataset; top N POS tags with high frequencies are selected.

IM-related features: Instant messages have their own special language styles. For example, popular internet words, special abbreviations, emoticons and words with similar pronunciation to others often appear in chatting text. Users select and use them according to theirs own language habits. We sum up the features of IM messages, and list some in Fig. 1.

Personal pronoun: 我,偶,咱,俺,你,伊,他,她
Modal particle：哈,呵,嘿,吼,嗬
Interrogative： 啥,呢,么,咋,麼,咦,吗
Expression: 寒,汗,晕,切,靠,囧,雷,赞,爽,哇,倒
Particle:额,呃,唔,啊,哦,嗯,恩,吧,啦,嗷,了,嘛,呀,滴
Emotion icons

Fig. 1. Part of IM related Features

Structural feature: in this paper we only choose one structural feature, the average length of sentences.

3.2   Model Generation

Theoretically we can use multiple classification models to solve authorship verification problem. However, in this problem the number of IM users is not fixed, we cannot determine the number of categories. For a user to be recognized, the data we can obtain is the user's own data (positive data), it is very difficult to collect enough negative samples that are representative of the entire negative class. One interesting problem for chatting text is that a person can use different chatting styles to chat with different peoples. For example, A has two friends B and C, A can chat with B using a totally different style as A chats with C. Usually if A decides a speaking style to chat with B, A will not change it; at least will keep it a relatively long time. That means we can not create a unique model to represent a user. Considering above special situations, one-class classification model is the best choice; in addition, a user need to build a recognition model for each of her/his chatting friends, for example, B and C will build a model of A respectively, and the two models may not be identical.

In this paper we choose two types of one-class classifiers. One is a feed-forward neural network with 'bottle-neck'[22]; the other is one-class SVM. Fig.1. shows the structure of the neural network. The neural network is trained using the standard back propagation algorithm to learn the identity function on the samples. The overall idea of the neural network is that while the bottleneck prevents learning the full identity function on m-space; the identity on the small set of samples is in fact learnable. Thus, the set of vectors for which the network acts as the identity function is a kind of sub-space which is similar to the trained set. Thus, the filter is defined by applying the network to a given vector; if the result is the identity, then the vector is "interesting".

To apply this idea to classify documents, we need to (i) decide on the number of hidden neurons and choose the appropriate learning rates. (ii) encode the documents as vectors, (iii) determine the appropriate acceptance thresholds when applying the trained networks to classify new documents.

If the documents are represented as an m-dimensional feature vectors, the structure of the network is m input units, m output units. By trying different network parameters, we set the parameters of the network as follows. the feed forward network has 3 layers, 37 input units, 37 output units and 18 hidden units. All units are standard sigmoid units. The learning rate is 0.75 and momentum coefficient is 0.08. The iteration for training is stopped until the mean-square error falls below a predetermined level. We set the threshold for classification according to the classification error of threshold selection set; the threshold should satisfy that the classification accuracy is about 90%. This means that we expect false negative is 10%.

In our work we use the LIBSVM available from http://www.csie.ntu.edu.tw/? cjlin/libsvm. This is an integrated tool for support vector classification. We use the standard parameters of the algorithm.
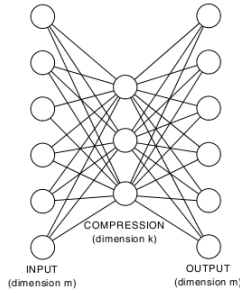
Fig. 2. A neural network with bottleneck

4   Experiment Result

The chat dataset used in this paper is collected from QQ, a widely used IM tool in China. The corpus contains three month's chatting text of different peoples. It includes 10 personal chatting logs and 14 group chatting logs of 2226 users in total. The images and the emoticon symbols are automatically transformed into text when we extract the chat logs from the QQ. Therefore, all these data are pure text. Information not relevant to the user, such as the system message, joke or notification copied from the web, is deleted.

Performance measure: we use the true positive rate (TP) and true negative rate (TN) rate to evaluate the performance of a classifier. TP is the ratio of the number of one user data correctly classified as this user to the total number of testing data of this user. TN is the ratio of the number of non-self data correctly classified as non-self to the total number of the non-self testing data.

Experiment setting: In the following experiments, we used 37 features in total: 16 function words, 12 token-based features, 9 IM related features. The data set for each user is partitioned into three sets for threefold cross-validation. In additional, we also build three shared sets; each includes 800 samples selected randomly from 14 group chatting logs, one for threshold selection, one for feature selection, and another for evaluating TN. We randomly select four users for training and testing. For the convenience of comparison, in the experiments we set the TN is about 90%, and then we compare the TP of each model. In practice we want a high TN; it means more pretenders are found.

Experiment 1: We use two methods to choose function words. Method 1(M1) only uses the training dataset of each user to select function words according their frequency. In method 2(M2), besides the user's training data set, one shared data set is used, the top 16 function words with high information gain in the training data set are chosen. The results are shown in table 1. (The number of training samples for each user is about 180, and the length of each sample is about 210).

Table 1. Performance of M1 and M2

User	M1(%)		M2(%)	
	TP	TN	TP	TN
U1	83	90	86	90
U2	87	90	91	90
U2	62	91	65	91
U3	62	89	60	89

Experiment 2: it is required for IM tools to recognize a user's identity in a short time. In reality it is also difficult to collect a large amount of corpus for a user. Thus, the length of data samples is a key factor for recognition. We chose different length of samples on the same training data set; the sample length is set as 120, 150, 180, 210, 240 and 300 Chinese characters, respectively. The corresponding numbers of training samples for each length are about 350, 250, 210, 180, 150, and 120, respectively. The results are shown in table2.

Table 2. Performance on the different sample size

Size	U1(%)		U2(%)		U3(%)		U4(%)	
	TP	TN	TP	TN	TP	TN	TP	TN
120	63	90	86	90	53	90	52	90
150	76	90	87	90	57	90	55	90
180	79	90	83	90	58	90	60	90
210	86	90	91	90	65	91	60	89
240	79	90	87	90	68	90	54	90
300	76	90	85	90	63	90	52	90

Experiment 3: We test the TP and TN using three type features separately, and then test the TP and TN using the combination of three types of features, the average results for the four users are shown in table 3. In table 3 F1 represents function words, F2 represents token based features, F3 represents IM-related features.

Table 3. Performance for Different Features

U	F1(%)		F2(%)		F3(%)		F1+F2+F3(%)	
	TP	TN	TP	TN	TP	TN	TP	TN
U1	87	90	74	90	62	90	90	90
U2	83	90	77	90	58	90	93	90
U3	62	91	60	91	55	91	71	91
U4	62	89	58	89	43	89	73	89

5   Conclusion

From the above experiments we can draw the following conclusions.

Conclusion from experiment 1: In general the performance of M2 is better than that of M1. For one-class classification problem, it is very likely that positive data and

negative data have many common features, in our experiment although the shared data set cannot represent the features of all negative samples; it is helpful to filter part of common features. This is the reason why we use two data sets (positive and negative) to select features.

Conclusion from experiment 2: Compared with a short sample, a long sample can more accurately represent the characteristics of instant messages. Therefore, with the increase of the sample's length, the performance of the one-class classifier is improved. However, for a fixed training data set, if the samples' length is too long, the number of training samples will be decreased greatly, this will affect the classifier's performance. The experiments show that when we set the sample size at 200 more or less, we get the best overall performance.

Conclusion from experiment 3:   The features we expect are the features that have big TP and TN. From the experiment result, function words have the best discrimination power, the IM related features also have the capability to recognize user identity, but their discriminative power is relatively weak, that is because most of the IM related features are popular among peoples, it is very difficult to select some as features of a specified person. By combining three types of features, we get better experimental results.

Here we only give the results of neural network. We also test the performance of one-class SVM; its TP is about 10 percent lower than neural network, so we do not show the experimental results of SVM.

References

1. Pavelec, D., Oliveira, L.S., Justino, E.J.R.: Using Conjunctions and Adverbs for Author Verification. Journal of UCS 14, 2967–2981 (2008)
2. Van Halteren, H.: Author Verification by Linguistic Profiling: An Exploration of the Parameter Space. ACM Transactions on Speech and Language Processing 4, 1–17 (2007)
3. Luyckx, K., Daelemans, W.: Authorship Attribution and Verification with Many Authors and Limited Data. In: 22nd International Conference on Computational Linguistics, pp. 513–520. ACL Press, Stroudsburg (2008)
4. Koppel, M., Schler, J.: Authorship Verification as a One-class Classification Problem. In: 21st International Conference on Machine Learning, pp. 1–7. ACM Press, New York (2004)
5. Abbasi, A., Chen, H.: Applying Authorship Analysis to Extremist-group Web Forum Messages. IEEE Intelligent Systems 20, 67–75 (2005)
6. Yule, G.U.: On Sentence-length as a Statistical Characteristic of Style in Prose: with Application to Two Cases of Disputed Authorship. Biometrika 30, 363–390 (1939)
7. Stamatatos, E.: Ensemble-based Author Identification Using Character N-grams. In: 3rd International Workshop on Text-based Information Retrieval, pp. 41–46. Springer, Heidelberg (2006)

8. Sanderson, C., Guenter, S.: Short Text Authorship Attribution via Sequence Kernels, Markov Chains and Author Unmasking: An investigation. In: 2006 International Conference on Empirical Methods in Natural Language Engineering, pp. 482–491. ACL Press, Stroudsburg (2006)

9. De Vel, O., Anderson, A., Corney, M.: Mining Email Content for Author Identification Forensics. SIGMOD Record 30, 55–64 (2001)

10. Forsyth, R.S., Holmes, D.I.: Feature Finding for Text Classification. Literary and Linguistic Computing 11, 163–174 (1996)

11. Zheng, R., Li, J., Chen, H., Huang, Z.A.: Framework for Authorship Identification of Online Messages: Writing Style Features and Classification Techniques. American Society of Information Science and Technology 57, 378–393 (2006)

12. Grieve, J.: Quantitative Authorship Attribution: An Evaluation of Techniques. Literary and Linguistic Computing 22, 251–270 (2007)

13. Keselj, V., Peng, F., Cercone, N., Thomas, C.: N-gram-Based Author Profiles for Authorship Attribution. In: 2003 Pacific Association for Computational Linguistics, pp. 255–264. Springer Press, Heidelberg (2003)

14. Kjell, B.: Discrimination of Authorship Using Visualization. Information Processing and Management 30, 141–150 (1994)

15. Argamon, S., Saric, M., Stein, S.: Style Mining of Electronic Messages for Multiple Authorship Discrimination: First results. In: 2003 ACM SIGKDD, pp. 475–480. ACM Press, New York (2003)

16. Argamon, S., Whitelaw, C., Chase, P., Hota, S.R., Garg, N., Levitan, S.: Stylistic Text Classification Using Functional Lexical Features. Journal of the American Society for Information Science and Technology 58, 802–822 (2007)

17. Zhao, Y., Zobel, J.: Effective and Scalable Authorship Attribution Using Function Words. In: Lee, G.G., Yamada, A., Meng, H., Myaeng, S.-H. (eds.) AIRS 2005. LNCS, vol. 3689, pp. 174–189. Springer, Heidelberg (2005)

18. Koppel, M., Schler, J.: Exploiting Stylistic Idiosyncrasies for Authorship Attribution. In: IJCAI 2003 Workshop on Computational Approaches to Style Analysis and Synthesis, pp. 69–72. AAAI Press, Menlo Park (2003)

19. Zhao, Y., Zobel, J.: Searching with Style: Authorship Attribution in Classic Literature. In: Thirtieth Australasian Computer Science Conference, pp. 59–68. Australian Computer Society Press, Darlinghurst (2007)

20. Kucukyilmaz, T., Cambazoglu, B.B., Aykanat, C., Can, J.: Chat Mining for Gender Prediction. In: Yakhno, T., Neuhold, E.J. (eds.) ADVIS 2006. LNCS, vol. 4243, pp. 274–283. Springer, Heidelberg (2006)

21. Kucukyilmaz, T., Cambazoglu, B.B., Aykanat, C., Can, F.: Chat Mining: Predicting User and Message Attributes in Computer-mediated Communication. Information Processing &Management 44, 1448–1466 (2008)

22. Manevitza, L., Yousef, M.: One-class Document Classification via Neural Networks. Neurocomputing 70, 1466–1481 (2007)

Using Hybrid Kernel Method
for Question Classification in CQA

Shixi Fan, Xiaolong Wang, Xuan Wang, and Xiaohong Yang

School of Computer Science, Harbin Insti-tute of Technology Shenzhen Graduate School,
518055, ShenZhen, China
`fanshixi@hit.edu.cn`, `{wangxuan,wangxl}@insun.hit.edu.cn`,
`yxh2008@gmail.com`

Abstract. A new question classification approach is presented for questions in CQA (Community Question and answering Systems). In CQA, most of the questions are non-factoid questions and can hardly be classified according to their answer types as factoid questions. A rough grained category is introduced and Multi-label classification method is used for question classification. That is, a question can belong to several categories instead of a specific one and the classification result is a category set. A two-step strategy is used for question Multi-label classification. In the first step, series binary classifiers of each question category are used separately. In the second step, results of those classifiers are combined and a set of question category is given as classification result. A hybrid kernel model, which combines tree kernel and polynomial kernel, is used for each binary classifier. A data set with 22000 questions is built and 20000 of which is used as training data, other 2000 as test data. Experiment result shows that the hybrid model is effective. A question paraphrase recognition experiment is carried on to verify the effectiveness of multi-label classification. The experiment results show that Multi-label classification is better than Single-label classification for questions in CQA.

Keywords: CQA, Kernel Method, Question Classification.

1 Introduction

Question classification is very important for QA systems, and the classification result directly affects the quality of the QA system. Firstly proposed by A.M. Turing (1950), QA is the task of automatically answering questions in natural language form [1]. Researchers have done much work on factoid questions analyzing and answering. However, most of the questions in real environment are non-factoid questions. Therefore, traditional QA systems cannot answer those non-factoid or complicated questions. The past few years saw the emergence of community-based QA system –CQA (e.g., Yahoo Answers[1], Live QnA[2], and Baidu Zhidao[3]). In CQA, people ask

---

[1] `http://answers.yahoo.com/`
[2] `http://liveqna.spaces.live.com/`
[3] `http://zhidao.baidu.com/`

questions freely, more importantly, other people may give answers according to their knowledge. Finally, the asker selects one best answer for the question. In this way, CQA groups people on Internet together and accumulates tremendous QA pairs which contain great knowledge. Analyzing and reusing those QA pairs are of great meanings [2]. One way is searching similar QA pairs according to question semantic similarity when a new question is asked. Classifying questions into some semantic category is helpful for question similarity calculating.

Question classification is a center problem for QA system. Question classification is helpful to indicate what a question is asking for and impose some constrains on the answers. Most of the previous researches focus mainly on factoid questions indicated by some interrogative keywords (when, where, who and etc) and answers are some short words. Machine learning methods are widely used for question classification problem. The information selected or features used are the most important factor for a classifier when a machine learning model is fixed. Sun (2008) extracts features from predict argument and parsing result [3]. WEN (2006) studies the feature extraction problem and uses subject predicating structure and interrogative words as basic feature set [4]. Hu (2008) applies POS tag and syntax structure features on answer classification [5]. Alessandro Moschitti (2007) encodes Predicate Argument Structures (PASs) into a svm for question classification [6]. All the approaches mentioned above work for factoid questions. However, most of the questions in CQA are non-factoid and the answers are no longer short words but a sentence or a few sentences. Thus, new question classification technique and strategy should be exploded. FAQFinder firstly uses machine learning methods for question classification in CQA [7]. Hyo-Jung (2005) proposes new question taxonomies for the encyclopedia service [8]. Duan (2008) find similar questions from CQA by Identifying Question Topic and Question Focus [9].

In this paper, we introduce a new question classification framework. The main difference between our work and previous study is that: (1) the question category is course grained and can cover any questions. (2) in our framework, a question may belong to several question categories according to its information need.

2   Question Classification Problem in CQA

2.1   Category for Question Classification

For factored questions, each question is assigned to one class which can be identified by some specific interrogative words. But in CQA, some questions are ambiguous and can be classified into several categories according to the information need. For example:

Will stock SZA000001 rise next week?

This question can be classified as yes/no question easily according to interrogative words. But in CQA, people will not prefer the answer like 'yes' or 'no' but the reason of rise or not for the stock. That is, a question can belong to several question types. This paper defines thirteen question types for questions in CQA. The question category is listed in table 1. The question category is rough grained and can cover most of the questions.

Table 1. Question category

Question type	Abbr.	Information need
Quantity	Wqua	The answer is measurement.
Description	Wdes	The answer need description
Yes/No	Wyes	The answer should be yes or no
Procedure	Wpro	The answer should be a series of event for something
Definition	Wdef	The answer is the definition of topic
Location	Wloc	The answer is location
Reason	Wrea	The answer can explain the question
Contrast	Wcon	The answer is the comparison of the items proposed in the question
Person	Wwho	The answer is about the people's information
Choice	Wcho	The answer is one of the choice proposed in the question
Time	Wtim	The answer is the data or time length about the event in the question
Entity	Went	The answer is the attribute of the topic.
Other	Woth	Other

2.2   Question Multi-label Classification

Now give the description of question Multi-label classification. Given a question category set $R = \{r_1, r_2, ..., r_n\}$, a question set Q, the single-label classification problem is to find a function $f : Q -> R$. In our framework, a question can belong to several types instead of a specific one and the classification result is a set of question category. Let $\Omega(R)$ denotes all the subset of R, and question classification can be defined as: $f : Q -> \Omega(R)$, obviously the Y space $\Omega(R)$ is very large, it is hard to find the function directly for data sparseness problem.

In this paper, a two-step method is presented for question Multi-label classification. In the first step, a series of binary classifiers are used for each question category:

$$C_i(q) = \begin{cases} r_i, if \quad q \quad \text{belong to } r_i \\ \overline{\quad 0, \quad \text{otherwise} \quad} \end{cases} : r_i \in R \tag{1}$$

Where $C_i(q)$ is used to judge that if or not question q belong to category C_i. In the second step, results of those classifiers are merged:

$$C(q) = \cup C_i(q) \tag{2}$$

Finally, $C(q)$ is given as the question classifying result.

3   The Hybrid Kernel Method for Question Multi-label Classification

There are many machine leaning models such as ME model, KNN and decision trees can be used for binary classification problem. However, all those models suffer the

same disadvantage: they cannot use the structure information of the question parsing result which is important for question classification. Researchers have developed tree kernels for this kind problem [10]. Generally, given an input space, a kernel-based method accomplishes its algorithm in a novel feature space derived from the input space. Thus, compared to ordinary nonlinear methods, kernel-based methods incur lower computational cost [11]. As a popular kernel-based method, Support Vector Machines (SVM) is linear functions of the form $f(x) = w \bullet x + b$ where $w \bullet x$ is the inner product between the weight vector w and the input vector x. The SVM can be used as a classifier by setting the class to 1 if $f(x) > 0$ and to -1 otherwise. Obviously, SVM can easily model features of Bag of words and Bag of POS tags by using polynomial kernels. However, the parsing tree structure features are also important for question classification. In this paper, we use both Polynomial kernels and Tree kernels for question classification problem.

3.1   Polynomial Kernels

For polynomial kernels, we use three kind features and those feature templates are listed as follows:

Word Features
Three kinds of word features are used: Unigram, Bigram and Trigram. The words are selected automatically according to their time frequency and discrimination. The discrimination is evaluated by the mutual information between the question type and word features.

Long Distance Rule Features
In questions, there are some fixed representation forms which can be treated as long distance rule features. For example:

   What is the difference between * and *

Here '*' can be any words sequence. 'What is the difference between' and 'and' are treated as long distance rule features. This kind rule based features can be identified by shallow string matching algorithm. Finally, 1105 long distance rule features are collected manually which can cover 61% questions statistically.

Special Interrogatives Words Features
Some interrogative words are ambiguous in semantic meanings when used in questions. To avoid this kind of ambiguity, more information is needed. The pre-word, post-word, pre-word POS tag and post-word POS tag are used as the context of the special interrogative words. The contexts together with the interrogative words are used as features. The number of special interrogative words is 35.

3.2   Tree Kernels

Our tree kernel method represents substructures of a parser tree as features. And the associated kernel function measures the similarity between two trees by counting the number of their common substructures. We use the subset tree (SST) as the substructures of a parser tree. Fig. 1 shows the parse tree of the sentence:
   " 如何计算收益 " together with its 6 SSTs (out of 17),
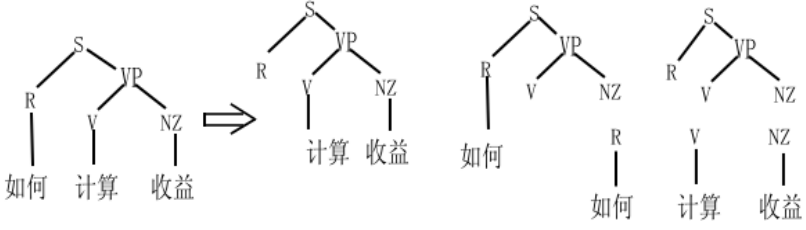
Fig. 1. A tree with some of its subset trees

Now give the main idea of tree kernel function which can calculates the number of the common substructures between two trees T1 and T2 without explicitly considering the whole fragment space. A tree kernel function can be defined as follows:

$$K(T_1, T_2) = \sum_{n_1 \in N_1} \sum_{n_2 \in N_2} \Delta(n_1, n_2) \tag{3}$$

Where $N1$ and $N2$ are the sets of all nodes in trees $T1$ and $T2$, respectively and $\Delta(n_1, n_2)$ is the number of common fragments rooted in the n_1 and n_2. $\Delta(n_1, n_2)$ can be calculated by an inductive procedure:

1 if the production rules at n_1 and n_2 are different then $\Delta(n_1, n_2) = 0$;

2 else if their children are the same and they are leave nodes, then $\Delta(n_1, n_2) = 1$;

3 else $\Delta(n_1, n_2) = \prod_{i=1}^{nc(n_1)} (\sigma + \Delta(c_{n1}^i, c_{n2}^i))$

Where $\sigma \in \{0, 1\}$ is a decay factor which in range [0, 1], $nc(n_1)$ is the children number of n_1 and c_{n1}^i is the i-th child of node n_1.

4 Experiment for Question Multi-label Classification

4.1 Data for Experiment

22000 questions are manually assigned question types according to the category defined in table 1. The statistics result for the 20000 training questions is shown in Fig. 2 and Fig. 3. Fig. 2 is the distribution of question types which shows that the top five question types (description, procedure, entity, yes/no and reason) cover most of the questions, in CQA. The top five question types are non-factoid question types and need reference or specific knowledge to answer. Fig.3 is the distribution of questions classified by type counts. 37% of the questions had one question type, 46% had two question types, 15% had three question types, 2% had more than four question types and the mean question type number is 1.82. The statistical result shows that questions with more than one question type is a common phenomenon and most of the questions have two question types wile few question has more than four question types.
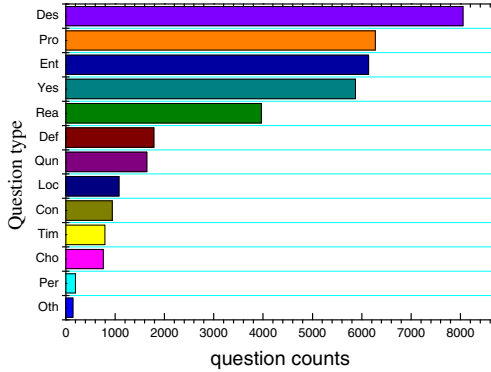
Fig. 2. Distribution of question types
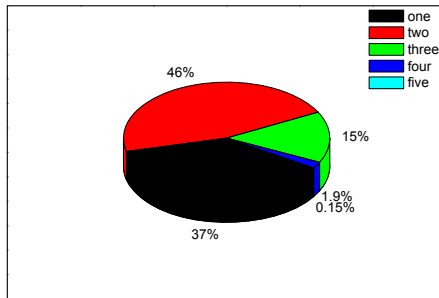
Fig. 3. Distribution of questions on type counts

4.2   Experiment for Question Multi-label Classification

In our question classification framework, we need a binary classifier. We use three models as binary classifier:

(1) ME: a Maximum Entropy model which uses the same features as polynomial kernels.
(2) PK: a SVM model which only uses polynomial kernels.
(3) TK: a SVM model which uses both polynomial kernels and tree kernels.

Two evaluations are used in this paper by considering human assigned label as measurement standard, one is to measure the precision ratio of binary classification, and the other one is to measure the precision ratio of question classification. Now give some basic definition:

N : The question number in test data.

P_i : The right classified number for i-th binary classifier.

P_q: The right classified question number. Note that only when the question classification result (the question category set) is identical to the real question category set.

The binary precision ratio can be defined as:

$$B - precision = \frac{\sum_{i \in C} P_i}{N * |C|} \tag{4}$$

The question precision ratio can be defined as:

$$Q - precision = \frac{P_q}{N} \tag{5}$$

Table 2 shows the performance of those three models. For question classification, TK use a hybrid kernel (tree kernel and polynomial kernel) which contains rich information than the other two models. TK achieves the best performance both in binary precision and question precision which is consistent to our assumption. The SVM model using polynomial kernel is better than ME model, although they have the same input features.

Table 2. Experiment result for question Multi-label classification

Question type	ME	PK	TK
B-precision	67.8	79.6	83.6
Q -precision	58.4	62.3	76.1

5   Question Paraphrase Recognition

To test the effectiveness of question Multi-label classification, a question paraphrase recognition experiment is carried. We setup two recognition algorithms, which use the result of question Multi-label classification and question single-label classification respectively. The experimental results show the power of question Multi-label classification.

Question paraphrase recognition is to judge if two questions have the same semantic meaning. Question paraphrase recognition is important for CQA. When people present a new question, it is convenient to list all the paraphrase questions which were presented earlier. 165 groups of question paraphrase set, which consists of 1082 questions, are selected from the CQA as test data. To test the effectiveness of QICA and multi-label classification, four methods are presented for question paraphrase recognition based on similarity calculation.

Let W(q) denotes all but integrative words of question q .

Let Ts(q) denotes our multi question type identification result, where Ts(q) is a question type set.

Let T(q) denotes the most probable question type of Ts(q). Obviously, T(q) is equivalent to a traditional question classification result.

The two question paraphrase recognition methods are defined as follows:

W(q)+ T(q)
This method uses words information and single question type of a question. For a question q, its paraphrase set can be defined as a set:

$$\Omega(q) = \{a : sim(a,q) > \delta, T(a) = T(q)\} \tag{3}$$

When the similarity between two questions (a and q) is higher than a specific value δ and their question type are identical, the two questions are treated as paraphrasing pair.

W(q)+ Ts(q)
The second method uses words information and our question type identification result which can be represented as:

$$\Omega(q) = \{a : sim(a,q) > \delta, \quad \exists t, t \in Ts(q) \& t \in Ts(a)\} \tag{4}$$

Eq. 4 indicates that if questions having a same question type they will be calculated.

$$
\begin{aligned}
sim(S_1, S_2) = &\; \alpha_1 * (1 - \frac{|L(S_1) - L(S_2)|}{L(S_1) + L(S_2)}) + \\
&\; \alpha_2 * (\frac{CSimW(S_1, S_2)}{L(S_1) + L(S_2) - CSimW(S_1, S_2)}) + \\
&\; \alpha_3 * (\frac{CSimP(S_1, S_2)}{L(S_1) + L(S_2) - CSimP(S_1, S_2)})
\end{aligned}
\tag{5}
$$

A comparing experiment between the four algorithms is done. Table 3 shows the performance of the two methods when the threshold parameters of δ is set to 0.75. Table 3 shows that the method using question Multi-label classification result achieved better performance of 63.7% on F1-measure. This means that our question Multi-label classification is better than traditional question classification of single question type.

Table 3. Question paraphrase recognition performance of different methods

method	Pre.(%)	Rec.(%)	F1(%)
W(q)+T(q)	60.3	46.6	52.6
W(q)+Ts(q)	58.3	70.3	63.7

According to Eq. 3, 4 the threshold δ has the directly relationship to the performance. Fig. 4 shows that with the increasing of δ, the precision increases and the recall decreases. When δ is in range [0.7, 0.9], these two methods achieve the best performance. Comparing W(q)+T(q) and W(q)+Ts(q), the previous method has a better performance on precision but suffer a bad performance on Recall. The F1-measure shows that methods using Ts(q) has a great advantage over methods using T(q). This result gives the proof that question Multi-label classification is better than

traditional single question classification for question paraphrase recognition. The question paraphrasing recognition verifies the effectiveness of our question classification approach.
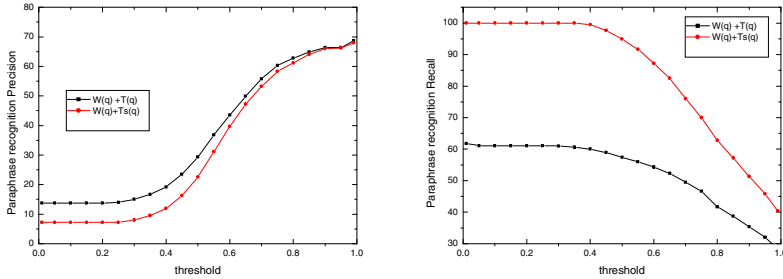
Fig. 4. Influence of threshold δ on question paraphrase recognition performance

6   Conclusion

In this paper, a new question classification approach for questions in CQA is presented. A two-step framework, which contains a series of binary classifiers, is used for question Multi-label classification. A hybrid kernel model is used as binary classifier. Experimental results show that the hybrid kernel model is better than ME model and SVM model with single kernel. The hybrid kernel model achieves 83.6% precision in binary classification and 76.1% precision in multi-label classification. The classification result is used for question paraphrase recognition. The experiment results show that algorithm using Multi-label classification result is better than the algorithm using single-label classification result.

References

1. Turing, A.M.: Computing Machinery and Intelligence. Mind 236(59), 433–460 (1950)
2. Liu, Y., Li, S., Cao, Y., Lin, C.-Y., Han, D., Yu, Y.: Understand-ing and Summarizing Answers in Community-Based Question Answering Services. Coling, 497–504 (2008)
3. Sun, A., Jiang, M.h., He, Y.f., Chen, L., Yuan, B.: Chine se Que stion Answering Ba sed on Syntax Analysis nd Answer Cla s sification. ACTA Electronica Sinica 36(5), 833–839 (2008)
4. Wen, X., Zhang, Y., Liu, T.: Syntactic Structure Parsing Based Chi-nese Question Classification. Journal of Chinese Information Processing 20(2), 33–39 (2006)
5. Hu, B.S., Wang, D.L., Yu, G., Ma, T.: An Answer Extraction Algo-rithm Based on Syntax Structure Feature Parsing and Classification. Chinese Journal of Computers 34(4), 662–676 (2008)
6. Moschitti, A., Quarteroni, S., Basili, R., Manandhar, S.: Exploiting Syntactic and Shallow Semantic Kernels for Question/Answer Classification. In: Proceedings of the Association of Computational Linguistics, pp. 776–783 (2007)

7. Tomuro, N.: Interrogative Reformulation Patterns and Acquisition of Question Pa-raphrases. In: Proceeding of the Second International Workshop on Paraphrasing, pp. 33–40 (2003)
8. Oh, H.-J., Lee, C.-H., Kim, H.-J., Jang, M.-G.: Descriptive Question Answering in Encyclopedia. In: Proceedings of the ACL Interactive Poster and Demonstration Sessions, pp. 21–24 (2005)
9. Duan, H., Cao, Y., Lin, C.-Y., Yu, Y.: Searching Questions by Iden-tifying Question Topic and Question Focus. ACL, 156–164 (2008)
10. Xu, Y., Zhang, D., Jin, Z., Li, M., YuYang, J.: Afast kernel-based nonlinear discriminate analysis for multi-class problems. Pattern Recognition, 1026–1033 (2006)
11. Gusfield, D.: Algorithms on Strings, Trees and Sequences. Cambridge University Press, Cambridge (1997)

Towards Understanding Spoken Tunisian Dialect

Marwa Graja, Maher Jaoua, and Lamia Hadrich Belguith

ANLP Research Group, Miracl Laboratory, University of Sfax, Tunisia
{marwa.graja,maher.jaoua,l.belguith}@fsegs.rnu.tn
http://sites.google.com/site/anlprg,
www.miracl.rnu.tn

Abstract. This paper presents a method for semantic interpretation designed for Tunisian dialect. Our method is based on lexical semantics to overcome the lack of resources for the studied dialect. This method is Ontology-based which allows exploiting the ontological concepts for semantic annotation and ontological relations for interpretation. This combination reduces inaccuracies and increases the rate of comprehension. This paper also details the process of building the Ontology used for annotation and interpretation of Tunisian dialect utterances in the context of speech understanding in dialogue systems.

Keywords: Tunisian dialect, Ontology, semantic annotation, semantic interpretation.

1   Introduction

The semantic parsing of understanding component in the context of dialogue systems helps to clarify the utterances meaning [1]. Since the spoken dialects are not written, it is very hard to obtain adequate corpora to deal with learning methods to understand utterances. In fact, approaches based on machine learning, still requires manual annotation of a large amount of training data. Moreover, tools of speech transcription are very expensive especially for Arabic dialect. On the other hand, using methods based on rules and parsers in the understanding module could pose inefficiency problem [2] especially in the case of a restricted domain. In fact, users often use keywords rather than well-structured sentences to request for information [3]. These characteristics are identified in a spoken dialogue for many restricted domain. Indeed, we noted the extensive use of keywords and the neglect of grammatical structures. These observations led us to use a lexical semantic approach to build the understanding module in Tunisian dialect. The method described in this work explores the meanings of words and their interconnection based on lexical choice databases. This is the subject of lexical semantics researches [4]. In this work, the lexical choice consists of integrating the domain and the task Ontology to allow annotation and semantic interpretation of utterances. Indeed, Ontologies have been used in several systems for semantic annotation. In this context, Allen and al. [5] use a domain independent Ontology to provide semantic representation in understanding module of a dialogue system. Also, the work of Milward [6] uses domain Ontologies to increase the level of clarification in dialogue manager.

Our contribution in this work consists in processing Tunisian dialect in dialogue systems. To our knowledge, this work is the first which deals with Tunisian dialect in an understanding component of a dialogue system. In this work we use domain Ontology to cover the lexicon used in the all services of railway station and we use a task Ontology which gathers all achievable tasks in this area as request information about the train time, booking, etc. To process utterances in Tunisian dialect in the restricted field, we exploit Ontologies for the semantic annotation and interpretation. Semantic annotation is used to assign semantic labels to each word without making a relationship between words or a group of words, while the semantic interpretation has a purpose larger than the semantic annotation. Indeed, the semantic interpretation increases the rate of comprehension by emphasizing the relationships between words in the same utterance.

This paper is organized as follows. The next two sections present an overview of Tunisian dialect and building Ontologies. Sections 4 and 5 present our method for the annotation and semantic interpretation based on Ontologies. The last tow sections show results and report conclusions.

2   Tunisian Dialect

Among variants of modern standard Arabic (MSA) are dialects which are spoken and informal. Arabic dialects differ from the MSA in terms of phonology, morphology, syntax and lexical level. Consequently, we could not use standard Arabic tools and adapt them to parse Arabic dialect. The Arabic dialect represents the real form of language. They are generally limited in use for informal everyday communication [7]. In fact, the dialect is mainly spoken and unwritten. So, it is crucial to study Arabic dialects in dialogue systems. We have chosen the Tunisian dialect as a representative example to study Arabic dialects in dialogue systems.

The Tunisian dialect is characterized by many features especially in the restricted field of railway services. To begin with, utterances in Tunisian dialect are not long. In fact, the average of word number in an utterance is about 3.61 [8]. Another feature in Tunisian dialect is the non respect of grammar and the use of foreign words especially in the studied field. In fact the most important key words are borrowings from French language. Finally, user utterances are characterized by a frequent use of domain dependant key words. All these features led us to investigate a method which focus on key word and relation between them and use a knowledge base to annotate and interpret utterances in Tunisian dialect by means of Ontologies.

3   Building Ontologies

Ontology is a formal specification, explicit and consensual conceptualization of a domain [9]. Indeed, the design and creation of Ontologies help humans to understand and solve the ambiguities for specific domains [10]. It consists of a set of concepts linked together in a methodological manner. In the literature, we can identify several types of Ontologies. In this work, we are interested in specialized Ontologies which are domain Ontology and task Ontology. Indeed, the domain Ontology provides

concepts of a domain [11] and relations between them. While task Ontology contains all tasks performed in a given domain [12]. It should be noted that the use of existing Ontologies is crucial. Nevertheless, we have not Ontologies in Tunisian dialect in the studied field. Therefore, we have manually built Ontology by following a known methodology. Several methodologies for building Ontology have been identified [9] [13]. We can mention as an example MethoOntology methodology and OntoClean methodology. There are other methodologies which are proposed for the construction of linguistic Ontology. TERMINAE methodology is one of the methodologies which allows manual construction of Ontologies from texts, based on language processing tools, to extract the lexicon and lexical or syntax relations.

The main purpose in this section is to explain the various steps followed in this work to build Ontologies using the TERMINAE methodology. To start the construction of our Ontology, we should have a corpus representing the domain knowledge. The used corpus is called *TuDiCoI* (*Tunisian Dialect Corpus interlocutor*) which is a corpus of spoken dialogue in Tunisian dialect. It is a pilot corpus which gathers a set of conversations recorded in the railway station between the staff and customers who request information about the departure time, price, booking, etc [8].

In an understanding module of a dialogue system, we annotate user utterances. That's why we process in this corpus only user utterances. So, among 369 as user utterances, we considered randomly 194 utterances as a development corpus which consists of 701 words. The rest of the utterances are left as a test corpus. We have manually transcribed the corpus because the lack of resources for automatic dialogue transcription especially for Tunisian dialect.

After fixing the corpus of a domain; we will continue the TERMINAE process to manually build the Ontology from a transcribed spoken corpus.

3.1 Lexical Specification

The lexical specification step consists of extracting the representative lexical of the domain and relations between lexical varieties. It is based on manual linguistic analysis of the corpus since we do not have language tools for Tunisian dialect. This step requires the intervention of linguistic experts to validate the obtained lexicon and lexical relations. After lexicon specification, we try to classify by group each set of semantic lexicon carrying the same semantic.

For example, {(3) لكسبراس ,(2) أسريع ,(1) نورمال } {*nwrmAl* (1), *Os~ryE* (2), *lksbrAs* (3)}[1]{*Normal* (1), *Rapid* (2), *Express* (3)} is a group which is carrying the same global semantic. It is the lexicon used to specify the train type.

3.2 Standardization

This step consists of assigning to each lexical variation a concept. So, from lexical varieties and lexical relations, we obtained a group of concepts and semantic relationships. At the end of this phase, we get a semantic network represented by a set of concepts linked by semantic relations. As example of standardization, the lexical

---

[1] For all examples, the transliteration is produced by the Buckwalter Arabic Transliteration System (http://www.qamus.org/transliteration.htm).

group {(3)لكسبراس ,(2) أسريع ,(1) نورمال} *{nwrmAl (1), Os~ryE (2), lksbrAs (3)}* *{Normal (1), Rapid (2), Express (3)}* is denoted by the concept "Train_Type".

3.3   Formalization

The purpose of the formalization step is to translate the semantic network obtained in the previous step into a knowledge representation language. In our work, the formalization is done by the OWL language (Ontology Web Language)[2]. OWL constitutes a knowledge representation language used to represent knowledge in a form usable by the machine.

4   Semantic Annotation

The semantic annotation is defined as the process used to associate semantic labels to each word or a group of words in a statement. In this work, we perform a semantic annotation of a transcribed speech based on domain Ontology and task Ontology. Hence, it is important to note that there are many problems in the utterances annotation since the dialect is the target of this study. In fact, we note that the spoken dialect does not respect correct grammatical form, which prevents us to use any analyzer of standard Arabic language and adapt it to the Tunisian dialect. The second problem is the segmentation of the utterance that appears as a key critic step in semantic annotation. Indeed, it sometimes becomes difficult to identify the text elements to annotate [11] because of morphological varieties of words, compound words, etc. So we have to make a standardization step of utterances before doing Ontology-based annotation by following the same standard used to build the Ontology in order to make a correspondence between elements of the Ontology with words of the utterance to be labeled.

− Treatment of compound words
Through the corpus study, we identified a definite list of compound words in the domain of railway information. This list is kept in a compound words dictionary. We have identified 55 significant compound words of the studied domain.

− Radicalization and lexical variants removal
The radicalization to remove lexical variants [14] is an effective method especially in the case of a limited field. Indeed, we try to deduce the singular form if the word is in the plural and the base form in the case of morphological variants.

After utterance normalization, we attribute a semantic label for each word in the utterance based on the domain Ontology and the task Ontology.   In fact, we exploit our Ontologies by scanning all concepts instances of both Ontologies and we look for the presence of a given word in the Ontology instances. It should be noted that it is possible to have two concepts, a one concept or no concept attributed to a given word. In case of having two different labels for a given word, it is possible to improve understanding through the phase of semantic interpretation which is presented in the next section. In case of a single semantic label for a word, we noticed a percentage of

---

[2] http://www.w3.org/TR/owl-features/

95% that the attributed label to a given word is correct. The case of non recognition of the semantic label of a given word is explained by incomplete words, a non-domain lexicon and peculiar phenomena to spontaneous speech which are not yet studied in this work. It can be noted in the example of figure 1 that the first word has two different semantic labels. So we should ameliorate semantic annotation by a semantic interpretation step to disambiguate the meaning.

The use of Ontology in the semantic annotation step does not provide significant benefit and its use in this step does not exceed the use of a domain dictionary. But the major contribution of the use of Ontologies is at the semantic interpretation presented in the next section. Therefore, the next step is a semantic interpretation which improves the semantic annotation by exploiting the semantic relations in Ontologies.

<div dir="rtl">

وقــتــاش يخرج الــتّران؟

</div>

wqtA\$ yxrj OtrAn

When does the train leave?

```
<token value='وقـتـاش '>
<annotation>Departure_Time_Request </annotation>
<annotation>Arrival_Time_Request </annotation>
</token>
<token value='يخرج '>
<annotation>Semantic_Relation</annotation>
</token>
<token value='الـتران '>
<annotation>Train</annotation>
</token>
```

Fig. 1. Example of semantic annotation

5   Semantic Interpretation

The semantic interpretation can be defined as a semantic decoding which clarifies the semantics carried in a statement and increases the accuracy of understanding. It is important to note that this work is not interested in contextual interpretation; it is only interested in lateral interpretation (i.e. context-free dialogue). Indeed, we try to improve labeling taking into account only semantic relations between the words of the same utterance and not an interpretation based on all utterances in the same dialogue.

The semantic interpretation allows exploiting linguistic links in an utterance and identifying their correspondence; as semantic relation; in the Ontology. So, it permits to link words together in the same utterance, so expressing a precise meaning.

The semantic interpretation is triggered for each word with two different labels detected after the semantic annotation step. Indeed, when we detect two labels for the same word, we traverse the utterance to detect the semantic relations in the utterance. If a relationship is identified in the utterance, we check in the Ontology if the target of this relationship is one of the concepts already identified as a label tag for a word or not. If yes, the target of the semantic relation is the correct label and should be attributed as a label to the word.

To explain the proposed method for semantic interpretation, we take the following utterance as an example: "لتونس إلماضي ساعه" "*ltwns IlmADy sAEh*" "*To Tunis One Hour o'clock*". First, we begin by the standardization step which consists of detecting the compound words and the semantic relationships. The statement becomes: "إلى تونس

إلماضي_ساعه" "*IlY twns IlmADy_sAEh*" "*To Tunis One Hour o'clock*". After the standardization step, we annotate semantically an utterance as it is presented in figure 2.

```
<token value='إلـﻯ '>
<annotation>Semantic_Relation</annotation>
</token>
<token value='تـونـس '>
<annotation>Arrival_City</annotation>
<annotation>Departure_City</annotation>
</token>
<token value='الـﻤـﺎضي_سﺎﻋه '>
<annotation>Hour</annotation>
</token>
```

Fig. 2. Example of semantic annotation

We note in figure 2 that the word "تونس" "*twns*" "*Tunis*" has two different semantic labels. It is labeled at the same time as "Departure_City" and "Arrival_City". Now, we apply the semantic interpretation step. To explain this point, we present an extract of our domain Ontology in a semantic network representation illustrated in figure 3. In this figure, the instance "تونس" "*Tunis*" belongs to two different concepts. At the interpretation level, we use semantic relations. In the utterance, there is the relationship "إلى" "*to*" which its target is the concept "مدينة_الوصول" "Arrival_City". So the word "تونس" "*Tunis*" should have as semantic label "مدينة_الوصول" "Arrival_City".
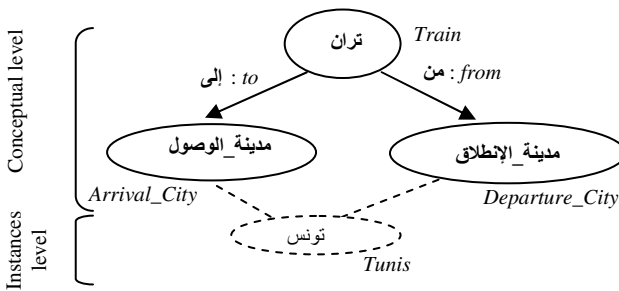
Fig. 3. Extract from the domain Ontology in a semantic network representation

6   Results

To build our Ontologies, we used the *Protégé* platform [3]. *Protégé* allows the generations of OWL file which represents the Ontology. In fact, OWL language is used in our work as a formalization language. To implement our method of semantic interpretation, we used the Jena framework. It is a Java framework for building

---

[3] http://protege.stanford.edu/

Semantic Web applications. The Jena Framework includes an OWL API[4] which facilitates the operating of the Ontology.

To test our method, we have used a test corpus which consists of 175 user utterances. This corpus contains 670 different words.

To evaluate the results of our method of semantic interpretation, we used Precision and F-measure. These measures are used to measure the semantic labels assigned to different words in oral utterances with relation to instances of Ontology concepts. In our case, the Precision measures the number of words correctly labeled divided by the total number of annotated words (correctly labeled and not correctly labeled) and F-measure measures the number of words correctly labeled divided by the total number of words of the test corpus.

Note that the number of words correctly recognized is 448, and the number of words which are not recognized is 208 words. So we get a Precision of 0.96 and an F-measure of 0.66.

Table 1. Experimentation results on the test corpus

Correct Annotation	*(a)*	448
Incorrect Annotation	*(b)*	14
Not Recognized	*(c)*	208
Total	*(d)*	670
F-Measure	*(a / d)*	0.66
Precision	*(a/(a+b))*	0.96

The precision ratio obtained for this evaluation is encouraging because we have not yet dealt with the specific phenomena of spontaneous speech. Indeed, after the analysis of 208 tokens which are not recognized in the interpretation, we noted that 38 tokens come from speech phenomena such as hesitation, incomplete words and the rest represent anaphors and out of vocabulary lexicon.

7   Conclusion

In this work, we have proposed a method which takes into account the specificity of the Tunisian dialect which has no linguistic processing resources. Indeed, the proposed method is based on lexical semantics which incorporates domain Ontology and task Ontology for the semantic interpretation of the utterance in a spoken dialogue without incorporating methods based on rules or parsers. In this method, we used the concepts of both Ontologies to annotate the utterances, while the semantic relations of Ontologies are used to disambiguate the interpretation and to increase the understanding level. To our knowledge, this method is the pioneer which proposes building domain Ontology and task Ontology for the Tunisian dialect in a limited domain. The proposed method is implemented and tested on a Tunisian dialect corpus using specialized framework in the Ontologies processing. Results are encouraging

---

[4] `http://www.openjena.org/`

for a first validation of this method. Indeed, we obtained an accuracy of 0.96. This precision is being improved by incorporating a processing level of particular phenomena of spontaneous speech.

References

1. Minker, W.: Stochastic versus rule-based speech understanding for information retrieval. Speech Communication, 223–247 (1998)
2. Sabouret, N., Mazuel, L.: Commande en langage naturel d'agents VDL. In: Proceedings of WACA 2005, pp. 53–62 (2005)
3. Mazuel L.: Utilisation des ontologies pour la modélisation logique d'une commande en langue naturelle. In: ColloRECITAL, Toulouse (2007)
4. Boite, R., Kunt, M.: Traitement de la parole Paris. Presses Polytechniques Romandes, France (1987)
5. Allen, J., Dzikovska, M., Manshadi, M., Swift, M.: Deep linguistic processing for spoken dialogue system. In: Proceedings of the Workshop on Deep Linguistic Processing, DeepLP 2007 (2007)
6. Milward, D., Beveridge, M.: Ontology-based dialogue systems. In: Proceedings of the 3rd Workshop on Knowledge and Reasoning in Practical Dialogue Systems (IJCAI 2003), Mexico (2003)
7. Diab, M., Habash, N.: Arabic Dialect Processing Tutorial. In: Proceedings of the Human Language Technology Conference of the North American, Rochester (2007)
8. Graja, M., Jaoua, M., Hadrich Belguith, L.: Lexical Study of A Spoken Dialogue Corpus in Tunisian Dialect. In: The International Arab Conference on Information Technology (ACIT 2010), Benghazi – Libya (2008)
9. Gruber, T.R.: A translation approach to portable ontology specifications. Knowledge Acquisition 5(2), 199–220 (1993)
10. Ghorbel, H., Bahri, A., Bouaziz, R.: Les langages de description des ontologies: RDF & OWL. In: Acte des huitièmes journées scientifique des jeunes chercheurs en Génie Electrique et Informatique (GEI 2008), Sousse, Tunis (2008)
11. Ma, Y., Audibert, L., Nazarenko, A.: Ontologies pour l'annotation sémantique, Ingenieries des connaissances (2009)
12. Mizoguchi, R., Vanwelkenhuysen, J., Ikeda, M.: Task Ontology for Reuse of Problem Solving Knowledge. In: Proceedings of the 2nd International Conference on Building and Sharing of Very Large-Scale Knowledge Bases, KB & KS 1995 (1995)
13. Guarino, N.: Formal ontologies and information systems. In: Proceedings of FOIS 1998. IOS Press, Amsterdam (1998)
14. Rijsbergen, C.J.V.: Information Retrieval. Butterworths, London (1979)

Topic Modeling of Chinese Language Using Character-Word Relations

Qi Zhao[1], Zengchang Qin[1,2], and Tao Wan[2]

[1] Intelligent Computing and Machine Learning Lab
School of Automation Science and Electrical Engineering
Beihang University, Beijing, China
[2] Robotics Institute, Carnegie Mellon University, USA
zhaoqi1861@gmail.com, zcqin@buaa.edu.cn

Abstract. Topic models are hierarchical Bayesian models for language modeling and document analysis. It has been well-used and achieved a lot of success in modeling English documents. However, unlike English and the majority of alphabetic languages, the basic structural unit of Chinese language is character instead of word, and Chinese words are written without spaces between them. Most previous research of using topic models for Chinese documents did not take the Chinese character-word relationship into consideration and simply take the Chinese word as the basic term of documents. In this paper, we propose a novel model to consider the character-word relation into topic modeling by placing an asymmetric prior on the topic-word distribution of the standard Latent Dirichlet Allocation (LDA) model. Compared to LDA, this model can improve performance in document classification especially when test data contains considerable number of Chinese words not appeared in training data.

Keywords: Topic Models, Latent Dirichlet Allocation, CWTM, Gibbs Sampler.

1 Introduction

Topic models are a class of hierarchical probabilistic models for analyzing discrete data collections. It assumes that documents are mixtures of topics and each topic is a probability distribution over words. Topic models have attracted a lot of attentions in recent years because it tries to model document in semantic level. Unlike English and the majority of alphabetic languages, the basic structural unit of Chinese language is character instead of word [13], and Chinese words are written without spaces between them. Most previous research applying topic models to analyze Chinese documents choose Chinese word as the basic term. Chinese documents are segmented into words which are generally believed to have more specific meanings than characters. However, word-based methods completely ignore the information that a Chinese word are composed of Chinese characters. Words sharing one same character may have some semantic relations, such a relation cannot be detected in word-based models. For example, the Chinese words "*xué xí* "(study) and "*xué shēng* " (student) are literally related by sharing the same character "*xué*", and they are also semantically related in meaning and may have a high probability to occur in the same context. However, in word-based computational models, these two words are treated as two distinct words has no relations at all.

The number of commonly used Chinese characters[1] is around 3000, while the size of word vocabulary can be way larger with new words created constantly. However, the characters constitute the new word probably already appeared in the history documents. In the standard Latent Dirichlet Allocation [3] model, all those words unseen in training data are assigned the equal probability in a topic by simply placing a symmetric dirichlet prior[8] on the topic-word distribution, regardless of their component characters having different occurrence rates in the training data. Hence, we will extend LDA by incorporating Chinese character-word relation to improve the performance of topic model when modeling Chinese documents.

This paper is structured as follows, the new proposed generative model is introduced in details in Section 2. We apply Gibbs sampling [1,5] method to inference the model in Section 3. Empirical results are given to evaluate this model in Section 4. The conclusions and future work is discussed in the end.

2   Character-Word Structure in Topic Modeling

2.1   Generative Model

To encode the character-word relation into topic model, we extend the standard LDA by placing an asymmetric prior on the topic-word distribution. This prior is obtained according to the character-word relation and topic-character distribution. The graphical model is shown in Fig. 1. Besides, we refer to this extended LDA model as *Character-Word Topic Model* (CWTM) in this paper.
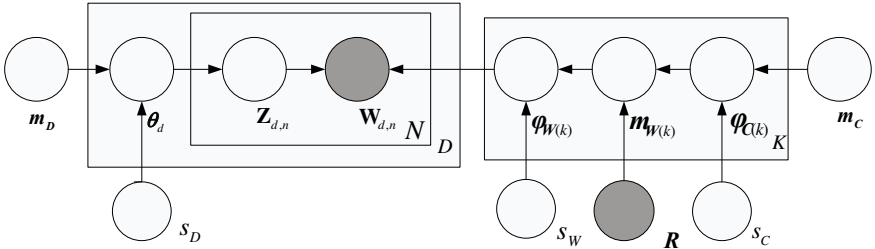
Fig. 1. Graphical model representation of Chinese character-word topic model

As illustrated in the Fig. 1, the document-topic modeling (the left part) is exactly the same as standard LDA [2,3]. $\mathbf{W}_{d,n}$ represents an observable Chinese word where d ($d = 1, \ldots, D$) is the index of document and n ($n = 1, \ldots N_d$) is the index of words in the document d. $\mathbf{Z}_{d,n}$ is a K-dimensional multinomial random variable indicate which topic is assigned to $\mathbf{W}_{d,n}$. Each document d is a mixture of topics parameterized by θ_d

---

[1] Previous research show that 3500 most commonly used characters can cover 99.48% of a corpus of written materials with over 2 million characters. 1000 most used character can cover 90% of a daily life corpus with over a billion characters [16].

which has a Dirichlet prior with hyperparameters m_D, s_D. The Dirichlet distribution is denote by $Dir(\mathbf{m}, s)$ in this paper, where $\mathbf{m}$ is the normalized *mean* ($\sum_i m_i = 1$) and s is called precision parameter (a scalar) that $\mathbf{m} * s$ is equivalent to α in standard Dirichelet distribution [8].

The CWTM model assumes that documents in a corpus $\mathbb{C}$ with Chinese word vocabulary size V_W and character vocabulary size V_C are generated by the following process:

1. For document $d = 1, \ldots, D$:
   (a) Draw a distribution over topics $\theta_d \sim Dir(\mathbf{m}_D, s_D)$.
   (b) For each Chinese words $n = 1, \ldots, N_d$, in document d:
      (i) Draw a topic assignment $\mathbf{Z}_{d,n} \sim Mult(\theta_d)$.
      (ii) Draw a word $\mathbf{W}_{d,n} \sim Mult(\varphi_{\mathbf{W}(\mathbf{Z}_{d,n})})$.
2. For each topic $k = 1, \ldots, K$:
   (a) Draw a topic distribution over Chinese characters $\varphi_{\mathbf{C}(k)} \sim Dir(\mathbf{m}_C, s_C)$.
   (b) Infer the distribution over Chinese words based on a deterministic function of the distribution of Chinese characters $\varphi_{C(k)}$ and character-word relationship $\mathbf{R}$: $\mathbf{m}_{\mathbf{W}(k)} = F(\varphi_{\mathbf{C}(k)}, \mathbf{R})$.
   (c) Draw a distribution over Chinese words $\varphi_{\mathbf{W}(k)} \sim Dir(\mathbf{m}_{\mathbf{W}(k)}, s_W)$.

$\mathbf{R}$ is a 2-dimensional matrix containing Chinese character-word composition relationship, it is defined as the count of character (C_j) contained in word (W_i):

$$\mathbf{R}_{ij} = Count(W_i, C_j) \quad i = 1, 2, ..., V_W \quad j = 1, 2, ..., V_C \tag{1}$$

where V_W and V_C represent the size of Chinese word vocabulary and Chinese character vocabulary, respectively. We define the deterministic function F as:

$$\mathbf{m}_{\mathbf{W}(k,i)} = F(\varphi_{C(k)}, \mathbf{R})_i = N_k(\prod_{j=1}^{V_C} \varphi_{C(k,j)}{}^{R_{ij}})^{\frac{1}{\prod_{j=1}^{V_C} R_{ij}}} \quad i = 1, 2, ..., V_W \tag{2}$$

where N_k is a constant to ensure all the coordinates of $\mathbf{m}_{\mathbf{W}(k)}$ sums to 1 and the character-word relation can be used in other form by simply altering the definition of function F.

2.2   Likelihood

The major difference between this model and LDA is the prior of the topic-word distribution $\varphi_{\mathbf{W}}$. In LDA, all the K distributions $\varphi_{\mathbf{W}(1)}, \varphi_{\mathbf{W}(2)}, ..., \varphi_{\mathbf{W}(K)}$ own a common symmetric dirichlet prior. While CWTW impose each topic-word distribution $\varphi_{\mathbf{W}k}$ with a unique asymmetric Dirichlet prior parameterized by $\mathbf{m}_{\mathbf{W}(k)}$ and s_W[11]. This means that each $\varphi_{\mathbf{W}(k)}$ has a corresponding prior $Dir(\mathbf{m}_{\mathbf{W}(k)}, s_W)$. And the mean of this Dirichlet prior $\mathbf{m}_{\mathbf{W}}$ is obtained by a deterministic function F, which takes character-word relation $\mathbf{R}$ and topic-character distribution $\varphi_{\mathbf{C}(k)}$ as inputs. Owing to the deterministic characteristic of generating $\varphi_{\mathbf{W}(k)}$ from R and $\varphi_{\mathbf{C}(k)}$, K topic-character distributions $\varphi_{\mathbf{C}(1)}, \varphi_{\mathbf{C}(2)}, ..., \varphi_{\mathbf{C}(K)}$ should be different to ensure that topic-word distributions $\varphi_{\mathbf{W}(1:K)}$ own priors with different mean parameters $\mathbf{m}_{\mathbf{W}(1:K)}$. Therefore, the character-word relation is incorporated as prior, which could make it less sensitive to errors caused by character-word relation, rather than is hardcoded into the model[9]. And the balance

between the prior originated from character-word relation and observed data could be controlled by adjusting the Dirichlet precision parameter s_W.

According to the description in Section 2.1, the joint distribution of all the variables given hyperparameters is:

$$p(\mathbf{W}_d, \mathbf{Z}_d, \theta_d, \varphi_\mathbf{W}, \varphi_\mathbf{C} | \mathbf{m}_D, s_D, \mathbf{m}_\mathbf{C}, s_C, s_W, R) = \prod_{n=1}^{N_d} p(\mathbf{W}_{d,n} | \varphi_{\mathbf{W}(Z_{d,n})}) p(Z_{d,n} | \theta_d) \qquad (3)$$

$$\cdot p(\theta_d | \mathbf{m}_D, s_D) \cdot \prod_{k=1}^{K} p(\varphi_{\mathbf{W}(k)} | s_W, \mathbf{m}_{\mathbf{W}(k)}) p(\mathbf{m}_{\mathbf{W}(k)} | \varphi_{C(k)}, \mathbf{R}) p(\varphi_{C(k)} | s_C, \mathbf{m}_C)$$

where $p(\mathbf{m}_{\mathbf{W}(k)} | \varphi_{C(k)}, \mathbf{R}) = 1$ only when $\mathbf{m}_{\mathbf{W}(k)} = F(\varphi_{C(k)}, \mathbf{R})$ and 0 for all the other values of $\mathbf{m}_{\mathbf{W}(k)}$ because F is a deterministic function.

3   Inference

Several methods have been proposed to do inference in LDA-like topic models [2,3]. In this paper, we will use Gibbs sampling [5], which is a special form of Markov chain Monte Carlo [1,4] for CWTM inference.

3.1   Document Likelihood

As we can see from the graphical model illustrated in Fig. 1, the joint distribution $p(\mathbf{W}, \mathbf{Z} | \mathbf{m}_D, s_D, \mathbf{m}_\mathbf{W}, s_W)$ could be factored as follows:

$$p(\mathbf{W}, \mathbf{Z} | \mathbf{m}_D, s_D, \mathbf{m}_\mathbf{W}, s_W) = p(\mathbf{W} | \mathbf{Z}, \mathbf{m}_\mathbf{W}, s_W) p(\mathbf{Z} | \mathbf{m}_D, s_D) \qquad (4)$$

And the two terms on the right side of above equation can be obtained by canceling out $\varphi_\mathbf{W}$ and θ respectively:

$$p(\mathbf{W} | \mathbf{Z}, \mathbf{m}_\mathbf{W}, s_W) = \int p(\mathbf{W} | \mathbf{Z}, \varphi_\mathbf{W}) p(\varphi_\mathbf{W} | \mathbf{m}_\mathbf{W}, s_W) d\mathbf{m}_\mathbf{W}$$
$$= \prod_{k=1}^{K} \frac{\Gamma(s_W)}{\Gamma(s_W + n_k^T)} \prod_{i}^{V_w} \frac{\Gamma(n_{k,i}^{TW} + s_W * \mathbf{m}_{\mathbf{W}(k,i)})}{\Gamma(s_W * \mathbf{m}_{\mathbf{W}(k,i)})} \qquad (5)$$

$$p(\mathbf{Z} | \mathbf{m}_\mathbf{D}, s_D) = \int p(\mathbf{Z} | \theta) p(\theta | \mathbf{m}_\mathbf{D}, s_D) d\theta$$
$$= \prod_{d=1}^{N} \frac{\Gamma(s_D)}{\Gamma(s_D + n_d^D)} \prod_{k=1}^{K} \frac{\Gamma(n_{d,k}^{DT} + s_D * \mathbf{m}_{D(k)})}{\Gamma(s_D * \mathbf{m}_{D(k)})} \qquad (6)$$

where n^{TW} is a matrices stored the counts of the number of times each Chinese word is assigned to each topic, say $n_{k,i}^{TW}$ denotes the count number of Chinese word indexed by i is assigned to topic k. Similarly, $n_{d,k}^{DT}$, element of n^{DT}, denotes the count number of Chinese words which are assigned to topic k in document d. And $n_k^T = \sum_{i=1}^{V_w} n_{k,i}^{TW}$, $n_d^D = \sum_{k=1}^{K} n_{d,k}^{DT}$.

3.2 Collapsed Sampler

To perform Gibbs sampling, we need a sampler $p(\mathbf{Z}_{d,n} = k|\mathbf{Z}_{d,n-}, \mathbf{W})$ from which new samples are drawn. This sampler is similar to that of standard LDA in [5]. The derivation of standard LDA's inference algorithm based on Gibbs sampling is introduced with details in [6].

$$p(\mathbf{Z}_{d,n} = k|\mathbf{Z}_{d,n-}, \mathbf{W}) \propto \frac{[n_{k,v-}^{TW} + \mathbf{m}_{\mathbf{W}(k,v)}][n_{d,k-}^{DT} + s_d * \mathbf{m}_{D(k)}]}{\sum_{i=1}^{V_w} n_{k,i-}^{TW} + s_W} \tag{7}$$

Noting $\mathbf{W} = \{\mathbf{W}_{d,n} = v, \mathbf{W}_{d,n-}\}$, the subscript ' $-$ ' represents current token indexed by (d, n) is not taken into consideration. Besides, $n_{k,i-}^{TW}$ denotes the count number of Chinese word indexed by i is assigned to topic k.

Since this Gibbs sampling method directly estimate $\mathbf{Z}_{d,n}$ for each word in the corpus, topic-word distributions and document-topic distributions can be obtained by:

$$\varphi_{\mathbf{W}(k,i)} = \frac{n_{k,i}^{TW} + s_W * \mathbf{m}_{\mathbf{W}(k,t)}}{\sum_{i=1}^{V_w} n_{k,i}^{TW} + s_W} \tag{8}$$

$$\theta_{d,k} = \frac{n_{d,k}^{DT} + s_D * m_{D(k)}}{\sum_{k=1}^{K} n_{d,k}^{DT} + s_D} \tag{9}$$

3.3 Estimate Topics over Chinese Characters

The Gibbs sampler proposed in Section 3.2 assumes $\mathbf{m}_{\mathbf{W}}$ as known, which is converted from topics over Chinese character through a deterministic process by the function $F(R, \varphi_{\mathbf{C}})$. Therefore it is necessary to estimate topics over Chinese characters $\varphi_{\mathbf{C}}$.

To approximately estimate $\varphi_{\mathbf{C}}$, we assume each Chinese character contained in the Chinese word token $\mathbf{W}_{d,n}$ in the corpus is directly sampled from the corresponding character-topic Multinomial distribution:

$$C_{d,n}^{(j)} \sim Mult(\varphi_{C(\mathbf{Z}_{d,n})}) \tag{10}$$

where j denotes the index of the character in the word token $\mathbf{W}_{d,n}$.

This means all characters in a word shared the identical topic, which is the same as the word's topic assignment. That is to say, if Chinese word "$jì\ suàn$"($\mathbf{W}_{d,n}$) is assigned with topic k, then its component characters "$jì$"($C_{d,n}^{(1)}$) and "$suàn$"($C_{d,n}^{(2)}$) are assigned the same topic k. This makes sense because characters in a word are inclined to express more related meanings.

Thus, the topic-character distribution could be approximately estimated as:

$$\varphi_{\mathbf{C}(k,j)} = \frac{n_{k,j}^{TC} + s_C * \mathbf{m}_{C(k,j)}}{\sum_{j=1}^{V_C} n_{k,j}^{TC} + s_C} \tag{11}$$

where $n_{k,j}^{TC}$ denotes the count number of Chinese character indexed by j is assigned to topic k.

The hyperparameters for CWTM are: $\mathbf{m}_D$, s_D, $\mathbf{m}_C$, s_C, s_W, which are either Dirichlet mean parameter or Dirichlet precision parameter. As with the standard LDA model, we use symmetric priors in the latter experiment section. That is to say, we set $\mathbf{m}_D$ and $\mathbf{m}_C$ as uniform distributions. For the other Dirichlet precision parameter, Griffiths have given a setting that show good quality in [5]. So we use that setting to fix $s_D = 50$, $s_W = V_W * 0.1$ and $s_C = V_C * 0.1$ for all k topics.

4   Experimental Studies

4.1   Data Descriptions

The Chinese corpus used in this paper is a news archive for classification provided by the Sogou laboratory[2]. The documents in this corpus are news articles collected from the website of the Sohu.com, which is one of the China's biggest Internet media company. These news articles are manually edited and classified into 10 classes that cover military, education, tourism and other topics. In the original corpus, each class contains 8000 documents. But few of these documents contain nothing but some meaningless non-Chinese symbols. Therefore those documents contain less than 100 Chinese characters will be ignored in our research. We then selected 1000 documents with 100 documents per class from the corpus as our experiment data which is named as NEWS1K and it is open to public online[3]. To take Chinese word as basic unit of topic model, we split Chinese documents into word tokens by using a Chinese word segmentation tool ICTCLAS-09[4] in our experiments. We also removed rare terms that appears less than 10 times across the whole corpus. For those terms that appears in over 50% of the documents, we consider them as stop words and remove them from the corpus as well. After the above preprocessing steps, the corpus NEWS1K contains 21389 unique Chinese words and 3631 unique Chinese characters. We then split NEWS1K into training and test set by the rate 1 : 1. There are 3747 Chinese words only appeared in the test set, but are unseen in training set.

4.2   Extracted Topics and Document Classification

The implementation of CWTM is based on GibbsLDA++[5]. Then we separately apply the standard LDA and CWTM to analyze NEWS1K's training data with topic number varies from 10 to 100. Fig. 2(a) shows top 20 terms of two topics independently extracted from CWTM and standard LDA when the number of topics is 50. As we can see from the results, topics extracted from these 2 models are relevant to sports.

As we mentioned in Section 1, for the words not appeared in training data, characters composing these words have appeared as component of other words in the training data. CWTM takes the character-word relation into consideration. Therefore, as is illustrated

---

[2] The corpus can be obtained at: http://www.sogou.com/labs/dl/c.html

[3] Dataset is available at: http://icml1.buaa.edu.cn/projects/topicmodel/index.html

[4] ICTCLAS is an integrate Chinese lexical analysis system which provides a tool for Chinese word segmentation. http://ictclas.org/

[5] C/C++ Implementation of LDA. http://gibbslda.sourceforge.net/

in Fig. 2(b), some of the words relevant to sports, even not appeared in the training data, were assigned relatively higher probabilities in topics about sports by CWTM. For example, the word "*zhǔ kè chǎng*"(home and away), though never appeared in training documents, its component characters "*zhǔ*", "*kè*", and "*chǎng*" are the components of those words appeared in training data. Therefore, it obtained a higher probability in CWTM. On the other hand, not surprisingly, none of these words were ranked into top 1000 words in the corresponding topic extracted by standard LDA model. Similar situations as described above happened with different number of topics.

Term Rank	CWTM (Topic 6)	Standard LDA (Topic 45)
1	队	队
2	比赛	比赛
3	场	场
4	球	分钟
5	球队	两
6	但	球
7	下	球队
8	胜	出
9	本	但
10	轮	下
11	主场	本
12	足	联赛
13	队	轮
14	比	被
15	队员	次
16	表现	表现
17	联赛	号
18	对手	主场
19	球迷	胜
20	冠军	球迷

(a) Top 20 words of extracted topics from CWTM and the standard LDA when the number of topics is 50.

Term Rank	CWTM (Topic 6)
450	主客场
453	德国队
468	大胜
477	发球
480	国奥队
532	接球
536	对攻
580	女足
592	赛道
783	国奥
859	必胜
863	落选
933	亚运会

(b) Some words, though not appeared in training data, ranked in top 1000 of the 6th topic extracted from CWTM with topic number equals to 50.

Fig. 2. Samples of extracted topics by using two different topic models: LDA and CWTM

The above extracted topics indicates that our method to incorporating character-word relation make the topics more reasonable. Then we make experiments to see its performance in document classification compared to standard LDA[12]. When modeling documents with topic model, each document can be represented as a distribution over topics by parameter θ. If two documents are semantically similar, it means the distance between two corresponding topic distributions is small, too. By using topic distributions, we can quantitatively measure the semantic (dis)similarities between documents. In such a way, documents of each class can be mapped to a K-dimensional space where K is the number of topics. In such a space, we can use discriminative machine learning algorithms, such as Neural Network (NN), k-Nearest Neighbor (kNN) or Support Vector Machine (SVM), to classify these documents based on the *semantic distance* measure in terms of dissimilarity between topic distributions θ. In particular, we employ the SVM to the classification task on the NEWS1K dataset for its effectiveness in text classifications such as question classification [10,7]. We adopted libsvm[6] with its default RBF kernel in our experiments.

---

[6] LIBSVM is an integrated software for support vector classification, regression and distribution estimation. http://www.csie.ntu.edu.tw/~cjlin/libsvm/
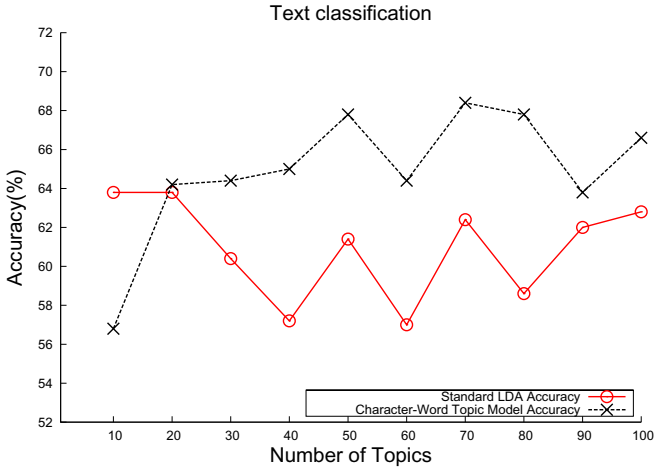
Fig. 3. Text classification accuracy based on standard LDA and CWTM

The classification accuracy of standard LDA and CWTM on the test data with different number of topics k is shown in Fig. 3. CWTM have a higher classification accuracy at all numbers of topic except when number of topic equals to 10. This may be caused by the too much errors brought to the topic-word distribution $\varphi_{\mathbf{W}}$ by the character-word relation prior when the number of topics is very small. Generally, our model have a better performance in document classification when a considerable number of words appeared only in test data while not in training data.

5   Conclusions

In this paper, we propose a method to incorporate character-word relation into the topic model by placing an asymmetric prior on the topic-word distribution of standard Latent Dirichlet Allocation (LDA) model. And experiments show, compared to LDA, CWTM can extract more reasonable topics and improve performance in document classification under certain circumstance. Besides, our proposed method can be easily applied to the most of the other topic models.

Though this method of encoding Chinese character-word relation could improve the performances of topic model, the errors brought to the model by this method prevent it from performing much better in modeling Chinese documents. This may be improved by altering the definition of the function $F(\varphi_{\mathbf{C}(k)}, \mathbf{R})$ or a more sophisticated model structure.

References

1. Bishop, M.C.: Pattern Recognition and Machine Learning. Springer, Heidelberg (2006)
2. Blei, D.M., Lafferty, J.D.: Topic Models. In: Srivastava, A., Sahami, M. (eds.) Text Mining: Classification, Clustering, and Applications. Data Mining and Knowledge Discovery Series. Chapman & Hall/CRC (2009)
3. Blei, D.M., Ng, A., Jordan, M.I.: Latent Dirichlet Allocation. Journal of Machine Learning Research 3, 993–1022 (2003)
4. Gilks, W.R., Richardson, S., Spiegelhalter, D.J.: Markov Chain Monte Carlo in Practice. Chapman & Hall, NewYork (1996)
5. Griffiths, T.L., Steyvers, M.: Finding Scientific Topics. Proceedings of the National Academy of Science 101, 5228–5235 (2004)
6. Heinrich, G.: Parameter estimation for text analysis. Technical report, Fraunhofer IGD (2009)
7. Huang, Z., Thint, M., Qin, Z.: Question Classification using Head Words and their Hypernyms. In: Proceedings of EMNLP, pp. 927–936 (2008)
8. Minka, T.: Estimating a Dirichlet distribution (2000)
9. Petterson, J., Smola, A., Caetano, T., Buntine, W., Narayanamurthy, S.: Word Feature for Latent Dirichlet Allocation. In: Proceedings of Neural Information Processing Systems (2010)
10. Qin, Z., Thint, M., Huang, Z.: Ranking Answers by Hierarchical Topic Models. In: Chien, B.-C., Hong, T.-P., Chen, S.-M., Ali, M. (eds.) IEA/AIE 2009. LNCS, vol. 5579, pp. 103–112. Springer, Heidelberg (2009)
11. Wallach, H., Mimno, D., McCallum, A.: Rethinking LDA: Why Priors Matter. In: Proceedings of the 23rd Annual Conference on Neural Information Processing Systems (2009)
12. Wu, Y., Ding, Y., Wang, X., Xu, J.: A comparative study of topic models for topic clustering of Chinese web news. In: Computer Science and Information Technology (ICCSIT), vol. 5, pp. 236–240 (2010)
13. Xu, T.Q.: Fundamental structural principles of Chinese semantic syntax in terms of Chinese Characters. Applied Linguistics 1, 3–13 (2001) (in Chinese)
14. Zhang, Y., Qin, Z.: A topic model of Observing Chinese Characters. In: Proceedings of the 2nd International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), pp. 7–10 (2010)
15. Zhao, Q., Qin, Z.: What is the Basic Semantic Unit of Chinese Language? A Computational Approach Based on Topic Models. In: Proceedings of Meeting on Mathematics of Language (MOL 2011), pp. 7–10 (to appear, 2011)
16. http://en.wikipedia.org/wiki/Chinese_language

Enrichment and Reductionism:
Two Approaches for Web Query Classification⋆

Ritesh Agrawal[1], Xiaofeng Yu[2], Irwin King[1,2], and Remi Zajac[1]

[1] AT&T Labs Research
201 Mission St. Ste 200
San Francisco, CA 94105
{ragrawal,irwin,remi}@research.att.com
http://www.research.att.com
[2] Department of Computer Science and Engineering
The Chinese University of Hong Kong
Shatin, NT, Hong Kong
{xfyu,king}@cse.cuhk.edu.hk
http://www.cse.cuhk.edu.hk/~{xfyu,king}

Abstract. Classifying web queries into predefined target categories, also known as *web query classification*, is important to improve search relevance and online advertising. Web queries are however typically short, ambiguous and in constant flux. Moreover, target categories often lack standard taxonomies and precise semantic descriptions. These challenges make the web query classification task a non-trivial problem. In this paper, we present two complementary approaches for the web query classification task. First is the *enrichment method* that uses the World Wide Web (WWW) to enrich target categories and further models the web query classification as a search problem. Our second approach, the *reductionist approach*, works by reducing web queries to few central tokens. We evaluate the two approaches based on few thousands human labeled local and non-local web queries. From our study, we find the two approaches to be complementary to each other as the reductionist approach exhibits high precision but low recall, whereas the enrichment method exhibits high recall but low precision.

Keywords: Query Classification, Unsupervised, Semi-supervised, Bayesian Approach.

1 Introduction

With the increasing popularity of search engines as the de-facto gateway to the World Wide Web (WWW), web queries have become an important medium by which a system can understand user's interests. Web queries can be however very diverse and any meaningful use requires classifying them into small commercial

---

taxonomy. Several challenges however make the task web query classification a difficult and a non-trivial problem. Web queries are typically short containing mostly two or three terms [1]. As a result, they tend to be ambiguous. For instance, the term "apple" can be either mean the fruit or the company or a gadget (apple computers). Web queries are also in constant flux and keeps on changing with current ongoing events (such as release of Apple's IPad or Japan Earthquake, nuclear leak, etc.). Moreover, target categories are not fixed, they depend on business requirements and often lack precise clean semantic descriptions.

As highlighted by the 2005 KDD Cup Challenge[1], the above challenges associated with the web query classification task has generated lot of interest in the academia and in the industry. Building on existing research in this area, this paper presents two complementary approaches for the web query classification task. First, the *enrichment method* uses the World Wide Web (WWW) to enrich categories and models the web query classification as a search problem. Second, the *reductionist approach* reduces a query to smaller subset of tokens that maintain the broad intention of the query. This smaller set of tokens are referred as central terms of a query, hence the reductionist approach here is is sometimes referred as the *centroid approach*.

That paper is organized as follows. In Section 2, we first explore some of the relevant work and existing approaches for web query classification. Section 3 presents the underlying theory and implementation of the proposed approaches. Section 4 discusses our evaluation strategy; we use crowdsourcing to obtain human labeled queries for training and testing purpose and further use standard measures of precision, recall and $F1$ for evaluation. Lastly, Section 5 presents conclusions and a discussion on the complementary nature of the two approaches.

2 Related Work

The task of web query classification is to classify queries into a set of predefined categories. Unlike document classification techniques, web query classification techniques have to deal with short queries and lack rich set of textual features, required for the classification purpose. To overcome the lack of rich query features, many researchers proposed query-enrichment based methods [2,3], also called post-retrieval techniques. Query-enrichment associates a collection of text documents to every query by sending the query to a commercial search engine and collecting the search engine results. Each query is represented by a pseudo-document bundling together the titles and snippets of the top ranked search result pages. These pseudo-documents are then classified into the target categories using text classification techniques. Since the target categories typically does not have associated training data, the KDD CUP 2005 winning solution solved the training problem by using the Open Directory Project (ODP) to build a an ODP-based classifier. The ODP taxonomy is then mapped to the target categories using various methods [4]. Thus, the post-retrieval query document is

---

first classified into the ODP taxonomy, and the classifications are then mapped into the target categories for web query classification. Using the above approach, the KDD cup winning solution [5] achieved an F1 measure of 0.44, which shows that accurate and robust query classification is still a difficult and open research problem.

Broder et al. [3] avoid the need for mapping between taxonomies (for instance from ODP to target categories) by using a set of keywords, attached to categories by human editor, as training documents. Although, their method achieves very good results ($F1=0.893$) on tails queries, it is difficult to compare these results to 2005 KDD Cup results as they use very different target taxonomy and dataset. Another challenge with their approach is that often target categories are just labels without any description of keywords.

Beitzel et al. [6] exploits both labeled and unlabeled training data for this task. Diemert and Vandelle [7] propose an unsupervised method based on automatically built concept graphs for query categorization. Some work has been dedicated to using very large query logs as a source of unlabeled data to aid in automatic query classification. Wen et al. [8] proposed a clustering method for query classification, which tried to associate related queries by clustering session data of query logs. The session data contain multiple queries and click-through information from users. Wen et al. [8] considered terms from result documents that a set of queries has in common. The use of query keywords together with session data has shown to be effective for query clustering. Beitzel et al. [9] tried to exploit some association rules between query terms to help query classification. Furthermore, they exploited several classification approaches, emphasized on an approach adapted from computational linguistics named selectional preferences, and used unlabeled query log data to mine these rules and validate the effectiveness of their approaches.

3   Web Query Classification Approaches

As shown in Eq. (1), the web query classification task can be modeled as the challenge of finding a category (c_i) that has maximum probability given a web query (q) as,

$$QC = \arg\max_c P(c_i|q). \tag{1}$$

In this paper, we use two complementary approaches to compute the probability of a category (c_i) given a query (q). Each of the two approaches are discussed in greater details in the following subsections.

3.1   Enrichment Approach

As discussed in Section 2, many researchers use the query-enrichment approach to overcome the short nature of a web query. A query-enrichment method works by transforming web queries into a set of pseduo-documents extracted from the WWW. In our enrichment approach, we use similar process but enrich target

OFFLINE

ONLINE
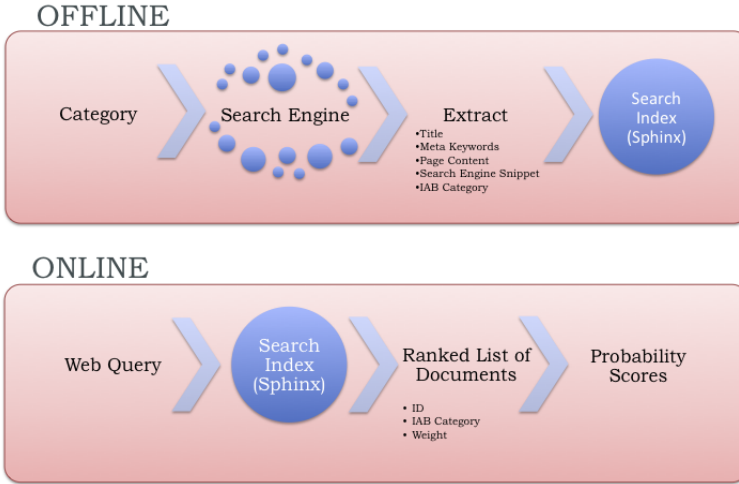
Fig. 1. Enrichment Method

categories instead of web queries. As shown in the Fig. 1, our category enrichment based approach is a two-step process.

Enriched Taxonomy. The first step, *an offline phase*, is the category enrichment process. Category enrichment is achieved by sending the category name as the search term to a commercial search engine and collecting the search engine results. For each category, we thus have a ranked list of documents where a document consists of a URL, a title, a search snippet, the URL's web page content, meta keywords and the category label that is used as a search term. The number of documents(γ) used per category is set empirically as described in Section 4.

Query Classification as Search. The second step, *an online phase*, is concerned with the actual web query classification task. Here we model the web query classification task as a search problem. Using Sphinx search engine[2], we first create a search index consisting of all the documents extracted in step 1. A web query that needs to be classified is then issued against this index. The results of this search includes a ranked list of indexed documents and associated BM25 relevance scores. Since, each document is assigned to one to more category from the offline phase, we can write the conditional probability of a category given a query as

$$P(c_i|q) = \sum_d P(c_i|d_j)P(d_j|q), \qquad (2)$$

where $P(c_i|d_j)$ is the conditional probability of a category (c_i) given a document (d_j) and $P(d_j|q)$ is the conditional probability of a document given a web

---

[2] http://sphinxsearch.com/

query (q). $P(d_j|q)$ is calculated as the normalized BM25 score of a document. For $P(c_i|d_j)$, one can naively assume it to be binary depending upon whether a document belongs to a given category or nor or, as discussed below, use a Bayesian transformation as shown in Eq. (3).

$$P(c_i|d_j) = \frac{P(d_j|c_i)P(c_i)}{P(d_j)}.$$ (3)

We calculate prior category probability from human labelled training data. $P(d_j|c_i)$ and $P(d_j)$ are computed using category names as search phrases against the above built index.

Using Eqs. (2) and (3), the enrichment method returns probability scores for all target categories for a given web query. In order to select relevant categories, we introduce two hyper-parameters, (1) threshold (α) and (2) number of categories (β), that are empirically tuned by running several tests on the training dataset. The threshold (α) is the minimum acceptable probability score of a category required to qualify as a relevant category. For instance, a threshold value of 15% indicates that only categories for which probability score is more than 15% are relevant. In some cases, this might still lead to too many categories. For instance, it is possible to have about 20 categories if the threshold is set to a low 5%. Since typically a search query belongs to a few categories, we also set a hard limit (β) on the number of categories that are selected. Thus, in the above example, if β is set to 3 then only top 3 categories, when ordered by decreasing probability scores, are selected.

3.2   Reductionist Approach

The hypothesis of our reductionist approach (or the centroid approach) is that, if queries share equivalent or synonymous centroid terms, these queries are very likely to share the same categories. Here, the centroid term in a query is the term that represents the broad and major intention of this query. In other words, this approach aims at reducing query terms (or tokens) to a smaller set of centroid terms while maintaining the broad intention of queries. As an illustrative example, consider the two queries "*harvard university*" and "*the london college*". The centroid terms of these queries are "*university*" and "*college*", respectively. Suppose we know that "*harvard university*" belongs to the category "*education*", it is very likely that "*the london college*" also belongs to the same category. Since they share synonymous centroid terms "*university*" and "*college*". If we have a reasonably large size of (e.g., several thousands of) queries with labeled categories, we can use them for classifying new queries.

We explore crowdsourcing [10,11] to obtain the labeled queries. Crowdsourcing describes outsourcing of tasks to a large group of people instead of assigning such tasks to an in-house employee or contractor, and allows to complete standard tasks more accurately in less time and at lower cost. Using crowdsourcing platforms such as Amazon Mechanical Turk, we distribute our query labelling
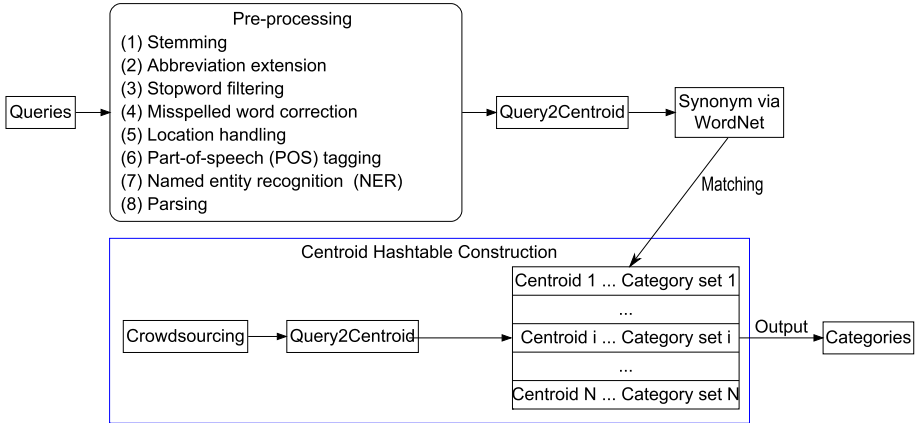
Fig. 2. Overall workflow for the centroid method

task to a large number of workers to obtain several labels per data point, and we apply statistical methods to filter out noisy labels. Consequently, we obtain approximately 3,300 labeled queries. We describe some details of our approach in Section 4.

Figure 2 shows the overall procedure. First, we perform some linguistics pre-processing of the query, including stemming, abbreviation extension, stopword filtering, misspelled word correction, location handling, part-of-speech (POS) tagging, named entity recognition (NER), etc. A number of off-the-shelf toolkits can be exploited for this purpose. For example, we use the porter stemming algorithm for query stemming, and we use Stanford POS tagger, NER tagger and parser for POS tagging, NER and parsing, respectively. The function Query2Centroid takes the pre-processed query as input, and identifies the centroid term of this query based on the POS tagging, NER and parsing results. For most queries, the centroid terms are nouns, verbs, noun or verb phrases. This considerably facilitates centroid identification, and we exploit rule-based methods for the Query2Centroid function, which performs effectively in practice. The function Synonym via WordNet returns all synonyms in WordNet for a query centroid term. For labeled queries, we conduct similar processing Query2Centroid to extract the centroid terms. We then construct a hash table in which the keys are centroid terms and the values are corresponding labeled categories. For an input query, we use synonyms of its centroid term to search the hash table to find the category. Note that the function Synonym via WordNet is useful since it enhances the coverage and boosts the recall. Take the query "*the london college*" for example to illustrate the procedure of the centroid method. After pre-processing, we list the POS tagging, NER and parsing results as follows:

POS tagging: `the/DT london/NNP college/NN`
NER: `the/O london/LOCATION college/O`
Parsing:
```
(ROOT
  (FRAG
    (NP (DT the) (NNP london) (NN college))))
det(college-3, the-1)
nn(college-3, London-2)
```

From these results, we know that "*london*" is a location name and college is a noun, and the function Query2Centroid can easily extract the centroid term "*college*" for the query "*the london college*". The function Synonym via WordNet finds all synonyms of "*college*" as "*college*" and "*university*". Suppose the hash table contains the key "*university*" and corresponding value (category) "*education*", we can easily and quickly find the category "*education*" for the query "*the london college*".

In summary, the centroid method is quite easy to implement as compared for example to related approaches mentioned in Section 2, while also performing reasonably well. The hash table look-up is very fast in on-line scenarios. This approach has a few limitations. First, as a reductionist approach, this method loses some information for query representation. Second, the quality of centroid identification (e.g., incorrect centroid terms) will affect the categorization performance. Third, for some queries, the categories cannot be found in the hash table. Therefore, this method exhibits high precision but low recall.

4   Experiment

Preparing Testing and Training Dataset. In order to test the two approaches, we extracted several thousands local and non-local web queries from search logs of two different commercially available search engines. A local search (such as "pizza near glendale ca", "walmart in new jersey city", etc.) is a specialization of the web (or non-local) search that allows users to submit geographical contained queries [12]. A non-lcoal search is any web search and ranges navigational searches (such as facebook, amzon api, etc) to informational searches (such as major stars in solar system, effects of global warming, etc.). Using Amazon Mechnical Turk's crowdsourcing system, each sampled query was labeled by 10 workers and assigned to one of the target category by each worker. In order to filter noisy data in the collected labelled data, we use two filtering steps. First, each worker is presented with a golden set of test queries randomly inserted into the labelling task. A golden set test query is a query for which we can easily and in an unambiguous manner identify the right category. For instance, one can easily say that "*Italian restaurant*" belongs to "*Food & Drink*" category given that there is no other category either related to Italian and restaurants. For each worker, we then calculate percentage adherence to these golden set queries by matching their selected categories to the expected target categories. We ignore all the labels from a worker who tends to differ from the golden set by more than 70%. This step removed about 3% labeled data.

Table 1. Performance of Enrichment and Centroid Method

	Local Search		Non-Local Search	
	Enrichment	**Centroid**	**Enrichment**	**Centroid**
Precision	0.613	0.616	0.428	0.707
Recall	0.421	0.409	0.281	0.233
$F1$	0.499	0.491	0.339	0.350

Second, for each web query, we calculate number of votes received across all the target categories and select only those categories that received more than 35% of votes. For instance, assume that out of 10 worker 4 say that the query "*financial news*" belongs to "*news*" category, while the other 4 say that it belongs to "*finance*" category and 2 say that it belongs to "*business*" category. In this particular, case we then accept "*news*" and "*finance*" as two correct categories and reject the "*business*" category. However, this constraint also introduces a limitation in our experiment. By restricting to only those categories that received more than 35% of votes, we are restricted to at-most two categories per query.

After this filtering, we are finally left with about 3,353 local and 3,324 non-local human labeled queries. Only 5% of queries in both the datasets have two categories associated with them. We randomly select 20% queries in the two datasets for testing purpose and use the rest 80% for training.

Results. As discussed in Section 3.1, the enrichment approach has three hyper-parameters, namely (1) threshold (α), (2) number of categories (β), and (3) number of documents (γ) used to represent a target category. In order to select optimal parameters, we run the enrichment algorithm several times on the training data with different parameter values. Figure 3 shows the influence of the three hyper-parameters on the $F1$ measure. In each of the figure, one parameter is varied while keeping the other two parameters constant. In Fig. 3 one notices the optimal performance of the enrichment method occurs by setting number of documents to 300, threshold to 15% and considering top two categories. We use the following settings to evaluate the performance of the enrichment method on the testing data.

Table 1 shows the precision, recall, and $F1$ measures for the two approaches on the local and non-local search queries. One notices that the performance of both the approaches (based on $F1$ Measure) significantly decreases for non-local searches as compared to local searches. This is partly because non-local searches tend to be much more diverse. For instance, a quick analysis of one month of local and non-local search log indicates that there are only 5% unique local searches in contrast to 23% unique non-local searches. This indicates that, as compared to local searches, non-local searches tend to be much more diverse and hence explains decreased performance of the two approaches on non-local searches.

Additionally, from Table 1 one also notices that the two approaches have comparable $F1$ measure; however, they demonstrate very different behavior. The centroid method exhibits higher precision as compared to the enrichment method, whereas the enrichment method shows higher recall as compared to the

Threshold = 0.15
No. Categories = 2
(a)

No. Documents = 300
No. Categories = 2
(b)

No. Documents = 300
Threshold = 0.15
(c)

variable
— Precision
···· Recall
---- F1
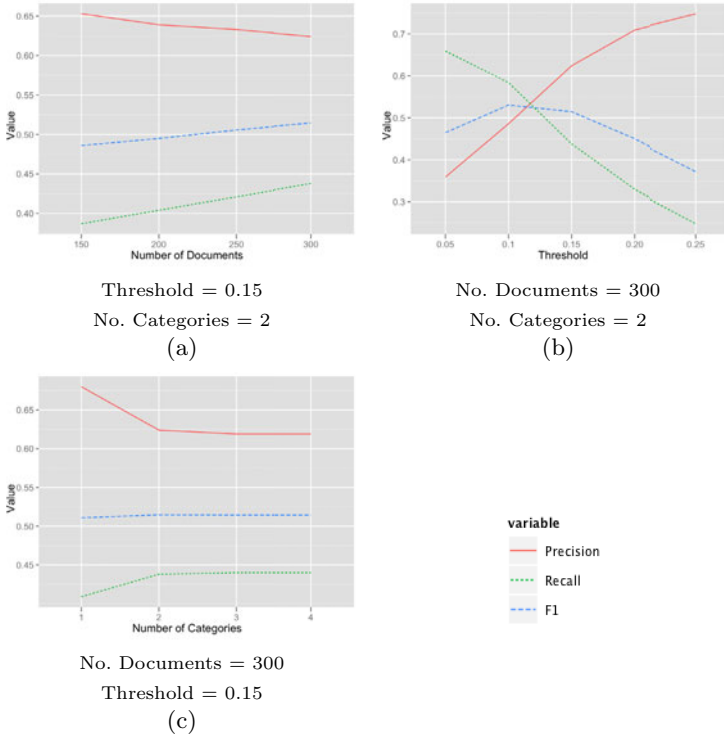
Fig. 3. Influence of hyper-parameters on the performance of the Enrichment method

centroid method. This is expected as the centroid method uses much more precise data (labeled queries) as compared to the enrichment data (online resources). As a result, the centroid method displays higher precision as compared to the enrichment method. On the other hand, however, using only labeled queries in part restricts the ability of the centroid method to deal with unseen centroid terms. In contrast, the noise in online documents helps improve recall of the enrichment method.

5   Conclusion

In this paper, we present two approaches–enrichment and reductionism, for the web query classification purpose. As demonstrated from the experiment, the two approaches are complementary in many different ways. First, on the theoretical level, the two methods approach the web query classification problem from two different ends. While the enrichment method focuses on enriching target categories, the reductionist approach focuses on reducing web queries to a few centroid terms. Second, at the pragmatic level, the two approaches demonstrate complementary precision and recall. The enrichment method has high recall but

low precision, while the centroid method has high precision but low recall. The complementary nature of the two approaches indicates a much higher performance can be achieved by combining the two using an ensemble technique [13], an aspect that we aim to further explore in future.

References

1. Baeza-Yates, R., Castillo, C.: Relating web structure and user search behavior. In: 10th World Wide Web Conference, Hong Kong, pp. 1–2 (2001)
2. Shen, D., Pan, R., Sun, J.-T., Pan, J.J., Wu, K., Yin, J., Yang, Q.: Query enrichment for web-query classification. ACM Trans. Inf. Syst. 24, 320–352 (2006)
3. Broder, A.Z., Fontoura, M., Gabrilovich, E., Joshi, A., Josifovski, V., Zhang, T.: Robust classification of rare queries using Web knowledge. In: Proceedings of the 30th annual international ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, pp. 231–238 (2007)
4. Shen, D., Sun, J.T., Yang, Q., Chen, Z.: Building bridges for Web query classification. In: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, WA, USA, pp. 131–138 (2006)
5. Shen, D., Pan, R., Sun, J.-T., Pan, J.J., Wu, K., Yin, J., Yang, Q.: Q2c@ust: our winning solution to query classification in kddcup 2005. ACM SIGKDD Explorations Newsletter 7, 100–110 (2005)
6. Beitzel, S.M., Jensen, E.C., Frieder, O., Grossman, D., Lewis, D.D., Chowdhury, A., Kolcz, A.: Automatic Web query classification using labeled and unlabeled training data. In: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Salvador, Brazil, pp. 581–582 (2005)
7. Diemert, E., Vandelle, G.: Unsupervised query categorization using automatically-built concept graphs. In: Proceedings of the 18th International Conference on World Wide Web, Madrid, Spain, pp. 461–470 (2009)
8. Wen, J.-R., Jian-Yun Nie, H.J.Z.: Query clustering using user logs. ACM Trans. Inf. Syst. 20(1), 59–81 (2002)
9. Beitzel, S.M., Jensen, E.C., Lewis, D.D., Chowdhury, A., Frieder, O.: Automatic classification of Web queries using very large unlabeled query logs. ACM Trans. Inf. 25(2) (April 2007)
10. Howe, J.: The rise of crowdsourcing. Wired 14(6) (2006)
11. Alonso, O., Lease, M.: Crowdsourcing 101: Putting the "wisdom of the crowd" to work for you. In: Tutorials of the 4th ACM International Conference on Web Search and Data Mining (2011)
12. Agrawal, R.J., Shanahan, J.G.: Location disambiguation in local searches using gradient boosted decision trees. In: Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS 2010, pp. 129–136. ACM, San Jose (2010)
13. Dzeroski, S., Zenko, B.: Is combining classifiers better than selecting the best one. In: Proceedings of the Nineteenth International Conference on Machine Learning, ICML 2002, pp. 123–130. Morgan Kaufmann Publishers Inc., San Francisco (2002)

Dynamic Template Based Online Event Detection

Dandan Wang, Qingcai Chen, Xiaolong Wang, and Jiacai Weng

MOS-MS Key lab of NLP & Speech,
Harbin Institute of Technology Shenzhen Graduate School, Shenzhen 518055, P.R. China
{wangdandanhit,qingcai.chen}@gmail.com,
wangxl@insun.hit.edu.cn, wjc_hit@foxmail.com

Abstract. In this paper, we propose a new dynamic template based event detection algorithm (DTED). Candidate template of an event is firstly constructed from a set of texts or their surrogates. Each candidate template contains several terms automatically extracted by the term weighting algorithm proposed in this paper. Then, we classify each text into a candidate event through a new similarity function. Some insignificant candidate templates are deleted. Whether an event template represents a new happened event or not is determined by comparing it with the event templates constructed in previous time window. Some events are merged into existing events and their templates are updated again. To evaluate the proposed DTED algorithm, we construct two datasets for experiment and F-measure is used as performance metric. The experiment result shows that DTED outperforms single-pass algorithm and clustering algorithms implemented in Cluto toolkit; meanwhile, Experimental results on Linguistic Data Consortium (LDC) dataset TDT4 show that DTED gets promising result.

Keywords: Dynamic template, online event detection, event detection and tracking.

1 Introduction

The booming of World Wide Web (WWW) greatly promotes the needs of fast detecting new happened events and comprehensively collecting news articles and weblogs (called "web documents" in this paper) about these events. They are very useful in applications like automatic editing and organizing web documents, intelligent news recommendation etc [1]. The large amount of RSS feeds provided by numerous websites is one important type of timely resource for document collection. RSS feeds precisely keep the chronological order of published articles and thus are very suitable for using in detecting and new happened events and tracking their evolution. Obviously, for the same event, there are usually a lot of web documents published in a relatively short duration (e.g. several hours). This duration is called a time window in this paper. For real applications, we can assume that most events will not develop within this duration. So web documents related to one event and published within this duration can be treated as describing the same event rather than any evolution of it. Under this assumption, we can use a batch of web documents for

real time topic detection and tracking (TDT), which is different from the input of traditional single-pass [4] based TDT techniques. In traditional TDT tasks, documents are processed one by one. Traditional TDT tasks also try to distinguish different topics about the same event. It is not necessary in the case that users prefer to get comprehensive viewpoints of an event within the same context. For this case, Wang et al (2008) proposes a method of clustering multiple topics into one news issues [2]. But the event detection mentioned above is different from news issues detection and other retrospective study of event since they usually put larger time granularity into consideration [3] [5].

Because that new events usually contain unknown terms that play important roles in events detection, it is difficult to construct event template only by existing terms and their statistical properties. So the named entities had been applied in most TDT techniques [9] [10]. In these techniques, named entities are emphasized through well designed term weighting frameworks and full documents are accounted in weighting procedure. Though useful in increasing clustering precision, this type of framework does not fully apply the merits of key terms of an event to improve both the detection efficiency and precision. When composing a news article or a weblog, the author usually pays a lot of effort on constructing its title to reflect its topic and to attract readers who concern the related event. This effort has been widely applied in commercial search engines, but it is not specifically treated in topic detection tasks.

In this paper, we divide the event detection task into two steps. In the first step, we construct event templates just by analyzing titles of web documents. In this step, important terms like named entities, verbs etc. are extracted from multiple document titles that are collected within the same duration. For each extracted term t, we partition the full document title set into two subsets, one contains document titles that include t, and left titles compose of another subset. As one document is usually called a story in TDT [4], we call the first subset a candidate story cluster. Candidate clusters are then sorted according to the number of titles they are composed of. Candidate clusters that just contain few stories are removed. According to the percentage of overlapped stories, a candidate cluster that is overlapped with another larger candidate cluster may be merged. The event template of a survived cluster is composed by several terms selected according to the weighting formula provided in this paper. The second step is classifying each document into event according to the similarity of document and event template. The full document text or abstract is involved in this step. Since similarity is computed between a document and an event template, a new similarity function is proposed to improve classification precision.

The rest of the paper is organized as follows: Section 2 presents our event detection algorithms. Section 3 describes the experimental data and evaluation metrics. The experimental results and discussions are given in Section 4. The paper is closed with conclusion and future work discussion in Section 5.

2   Dynamic Template Based Event Detection

2.1   Pre-processing of Web Document

The RSS feeds provide a very suitable news source for online event detection applications. But they usually do not contain full documents. To be comparable with existing text clustering algorithms, we use web pages crawled by spiders to conduct

our experiments. For the test purpose, titles, contents and important metadata such as publishing date, category, source, images, etc. are extracted from web pages. The word segmentation is performed for Chinese texts. And then we perform part-of-speech tagging and remove stopwords [11]. Some special noise words related to our corpus such as "新浪" (Sina), "博客" (blog), "转载" (reprint), "出处" (source) etc. are also treated as stopwords. Following the conceptions defined by Allan et al, we call a web page content with its title a story [8].

Though time window has been applied since Yang et al in retrospective research on events [4], we use it here for a little different purpose. We assume that the chronological order of web documents published within a time window t is not distinguishable. By this assumption, we divide story stream into windows with a fixed duration. The length of time window is set according to requirement of applications and the speed of story collection. It could be several minutes, hours or days.

2.2   General Framework of DTED Algorithm

Processing of DTED Algorithm: After above preprocessing of web documents, this paper divides the task into two steps: 1) candidate template of an event is constructed from a set of story titles; 2) classify each story into events through a new similarity function. Here the event template is defined as follows:

Definition 1. The *n*-order template of an event is a finite set of *n* (*term*, *weight*) pairs sorted in the descending order of weight values.

By above definition, each candidate template contains several terms automatically extracted by the term weighting algorithm proposed in latter section. Fig 1 and 2 illustrate the procedure of building event templates. At first, important terms such as named entities, verbs etc. are extracted from titles of stories in story set S_t. For each extracted term w, we partition S_t into two subsets $S_{t,r}$ and $S_{t,nr}$. The subset $S_{t,r}$ includes stories that contain term w in their titles and is thought as a candidate event cluster. Fig 1 illustrates a simple example of this procedure. In this figure, the story set $S_t = \{D_1, D_2, D_3, ..., D_7\}$, and totally 8 important terms are extracted from titles of stories in S_t. Then 8 clusters $\{c_1, c_2, c_3, ..., c_8\}$ are constructed corresponding to these terms. In this example, $c_1 = \{D_2, D_3, D_4, D_6\}$ and $c_6 = \{D_3, D_4, D_5\}$. It is obvious that these clusters may overlap with each other, e.g. c_1 and c_6 contain the same story set $\{D_3, D_4\}$.
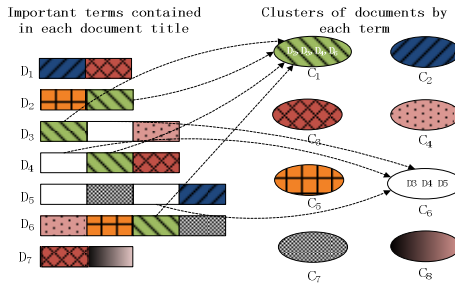
Fig. 1. Cluster documents by title terms

To detect new events and construct their templates, all candidate clusters are sort in the descending order of their cardinalities. Then a smaller candidate cluster may be merged into a larger one if their overlapping degree denoted as γ is greater enough. Those candidate clusters with cardinalities smaller than a threshold value are thought as noise and are removed from the candidate list. Each survived cluster represents a new detected event and is used to build template for the new detected events. This procedure is illustrated in Fig 2 for the same example as in Fig 1. Here clusters c_6, c_5, c_7, c_4 are merged into c_1 and clusters C_7, C_8 are removed for they contain no more than 2 stories. Clusters c_1 and c_3 are finally survived and two events e_1 and e_2 corresponding to these clusters are detected. The last task of the first step is building the event template. To finish this task, terms included in the story titles of an event cluster are weighted and ranked by their weights. Top n terms with their weights are then extracted to compose of the n-order event template. Though stories are clustered into some candidate event clusters in the first step, this is not the final decision. It is because that only title of each story is processed.
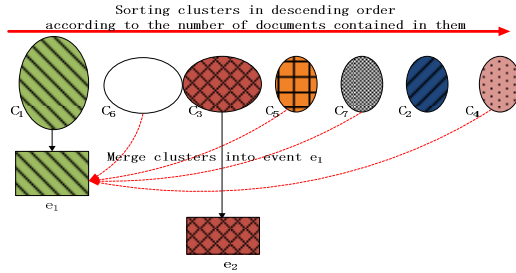
Fig. 2. Detect event from title clusters

Term Extraction and Weighting of DTED Algorithm: In this section, two questions that had been originated in the general DTED framework are answered: 1) How to extract important terms before building candidate clusters for them? 2) How to weight terms within a survived cluster for the construction of event template?

For the first question, we simply apply the incremental *idf* approach applied in [3] with minor variation in equation (1).

$$idf_t(w) = \log_2((M_t + 1)/(m_t(w) + 0.5)) \tag{1}$$

Where M_t is the cardinality of S_t, $m_t(w)$ is the number of stories that include term w in their titles. The term with its incremental idf_t greater than a threshold value α is the important term in our clustering step.

For answering the second question, we provide a weighting formula for term W in cluster C_i of tth time window in equation (2).

$$weight(w, c_i, t) = (df_t(w)/|c_i|) \cdot idf_t^t(w) \cdot \sum_{s \in c_i} tf_s(w) \tag{2}$$

where S is a story, $tf_s(w)$ is the term frequency of W in S, $df_t(w)$ is the document frequency of term W in cluster C_i, $idf_t'(w)$ is computed as in equation (1) except that the full story content is put into consideration rather than just processes the story title, $|C_i|$ is the cardinality of C_i. The n-order event template vector $\mathbf{V}_i$ of a cluster C_i is then constructed by ranking terms in descending order of their weights and choosing top n terms. In this paper, a normalization version of the original weight value calculated by equation (2) is stored into the event template. The normalization of a term weight value is computed as:

$$\overline{nw_m} = \frac{weight(w_m, c_i, t)}{\sum_m weight(w_m, c_i, t)} \tag{3}$$

Then the event template vector is represented by equation (4).

$$\mathbf{V}_i = \{(w_1, \overline{nw_1}), (w_2, \overline{nw_2}), ..., (w_n, \overline{nw_n})\} \tag{4}$$

Though for the n-order event template, usually n terms should be extracted from the event cluster, but sometimes, terms ranked near n may get very small weight values and putting them into the event template may cause a lot of noises. In such cases, these terms are removed from the template. In our model, the minimum weight value required for the event template is denoted as β.

Here we take the news event "US President Barack Obama's first visit to China" as a real example. Table 1 is the 15-order event template for this event. In the template, the top 3 terms "奥巴马" (Obama), "美国" (US), "中国" (China) label the main contour of this event, and other terms respect to various aspects of this event that people may concern about.

Table 1. An example of 15-order event template

Term	Weight	Term	Weight
奥巴马 Obama	0.3380	问题 Problems	0.0252
美国 US	0.2107	访华 Visit to China	0.0182
中国 China	0.0802	表示 Express	0.0182
中美 China-America	0.0631	访问 Visit	0.0144
总统 President	0.0484	战略 Strategy	0.0132
贸易 Trade	0.0318	亚洲 Asia	0.0120
关系 Relationship	0.0302	合作 Corporation	0.0106
经济 Economic	0.0287		

2.3   Similarity Function

For the comparison of the story and the event template, a suitable similarity measure is required. Though traditional Cosine function or other kind of similarity function is applicable for this task, we provide a variant of the Cosine similarity as follows.

$$sim(s, v_i) = \sum_{w \in s \text{ and } \mathbf{v}_i} (sw_{w,s} \cdot \overline{nw}_{w,v_i}) \cdot \sum_{w \in s \text{ and } \mathbf{v}_i} (score(w)) \tag{5}$$

$$score(w) = \frac{idf_t'(w)}{(\sum_{k=1}^{n} idf_t'(w_k)) / n} \tag{6}$$

Where w is the term belong to $\mathbf{V}_i$, $sw_{w,s}$ is the weight that term w supports the story s which is computed by the traditional *tf-idf* formula, $\overline{nw}_{w,v_i}$ is the normalized weight of term w in $\mathbf{V}_i$. $score(w)$ is to emphasize the relative *idf* value of a term in the event template, n is the order of $\mathbf{V}_i$.

3　Experimental Setup

3.1　Datasets and Evaluation Metrics

Self-Collected Datasets: Two Chinese datasets constructed from real web environment and they are used to evaluate the proposed DTED algorithm. Recall, Precision and F1-measure are applied for performance evaluation [7]. Topic annotation is done on the two datasets as the same as TDT annotation[1].

Dataset1 is crawled from the link http://news.sina.com.cn/zt/, which contains 13252 news pages that include picture pages, video pages, weblogs and flash videos etc. For those WebPages that contain non-text contents, we just extract their text contents or use their titles if no text contained in them. These pages are related to 28 events tagged by the website editors. The maximum event cluster contains 3122 stories, while the minimum event cluster has 19 stories.

Dataset2 is crawled from the Sina Blog website indexed by the hyper link http://blog.sina.com.cn/lm/zt/index.shtml. It consists of 1589 weblogs related to 40 events labeled by Blog site editors. The maximum event cluster has 138 weblogs, and the minimum event cluster contains just 4 weblogs. As mentioned by Wang and Zhang et al (2008), weblogs usually have much more noises than web news. So we set this experiment to compare its performance with news articles [2].

Standard Dataset: Experiments are also performed on standard Linguistic Data Consortium (LDC) dataset TDT4. It contains new stories from October 2000 to January 2001. We only use Mandarin stories which contain about 27145 documents reigning from news agencies such as Xinhua, ZBN, CTV, etc. All of the topics belong to one of the "13 Rules of Interpretation". In the evaluation of dataset TDT4, classical measure C_{Det} is used [4] [10].

4　Experiments and Discussion

4.1　Events Detection Results

The parameters mentioned in the paper for DTED are: $\alpha = 1.65$, $\beta = 0.01$, $\lambda = 0.6$, $\eta = 0.16$. As shown in Fig 3, the DTED algorithm reaches the best performance on

---

[1] http://www.ldc.upenn.edu/

Dataset1, of which the precision result 95.04% and F1-measure is 90.17%. Agglomerative hierarchal clustering reaches the highest recall rate and worst precision rate while its F1-measure result is nearly the same as single-pass algorithm. This result dose not agrees with the result in [3]. The direct-I2 and rbr-H2 algorithms implemented in Cluto toolkit[2] perform well in precision and perform poor in recall.
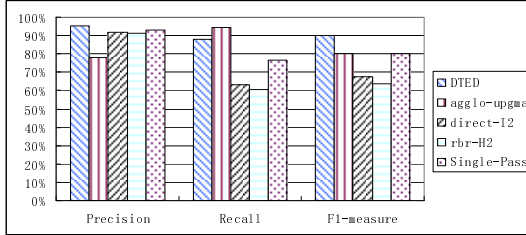
Fig. 3. Experiment results for Dataset1

Fig 4 shows the experiment results on Dataset 2. Be compared with Fig 3 we can see that the overall results of DTED and agglo-upgma algorithms on Dataset2 are worse than results on Dataset1. And the DTED also get the best performance with a little advantaged than Single-Pass and direct-I2. On this dataset, Single-Pass reaches the highest precision of 97.16%. But its recall result is the worst one.
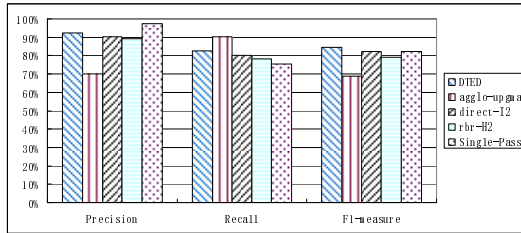
Fig. 4. Experiment results for Dataset2

We ran our system on the TDT4 dataset. Fig 5 shows the DET curve. The topic-weighted minimum normalized cost is 0.418 at a false alarm rate of 2% and a miss rate of 32%.

4.2   DTED Algorithm Discussion

In DTED algorithm, two critical factors are the event template vector and the event-story similarity function. In equation (2), we introduce the factor $df_t(w)/|c_i|$ (denoted as CF) on traditional *tf-idf* weighting. To see the effectiveness of CF and our similarity function given by equation (5), we conduct an experiment to compare the

---

[2] http://www.cs.umn.edu/~karypis/cluto
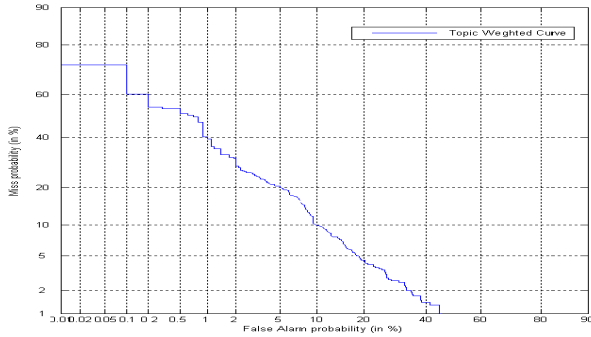
Fig. 5. DET curve on TDT4

DTED similarity function performance and traditional Cosine similarity function with and without CF for the term weighting (denoted as CF-DTED, nonCF-DTED, CF-Cosine, and nonCF-Cosine respectively). Results are shown in Fig 6. Here CF-DTED obviously outperforms other configurations. We argue that the Cosine function is not suitable here since our event template vector is only composed of several terms rather than centroid vector of all stories in the event cluster.
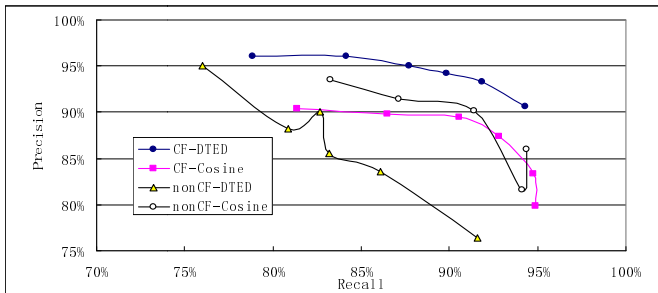
Fig. 6. Comparison of DTED similarity function and Cosine function on Dataset1

4.3   Online System for Large Scale Corpus Building

Though the Linguistic Data Consortium provides a series of TDT corpora for TDT algorithms evaluation[1], they are not suitable for us since we are trying to study the effectiveness of real web document titles on the event detection efficiency and precision. To facilitate the research of online event detection and tracking, we construct online event detection and tracking system based on DTED algorithm. It has continually run about 230 days to construct a large scale Chinese TDT corpus. Up to now, it had processed 1,258,870 weblogs or news titles and abstracts (some of them contain full texts) and had detected 1,452 important events with 103,519 related articles. We selected the 687 events detected during May 23, 2009 to September 1, 2009 and manually labeled the latest (rather than top ranked) 20 articles of each event

for their relevance. The result shows that the system reached the average precision of 84% and is applicable in real applications. Since the scale of the corpus is still increasing, it will be a very useful corpus for online event detection and tracing.

5   Conclusion and Future Work

In this paper, we propose a new two-steps DTED algorithm for online event detection. At first, the candidate template of an event is constructed from a set of story titles. Then each story is classified into events through a new similarity function. Since the clustering is just taken on document titles, DTED saved a lot of time for the detection task. Another merit of DTED algorithm is that the title-based event template takes the advantages of authors' effort on representing an event by few terms and thus improves the effectiveness of event representing model. Since the event template is represented just by several terms, which saves the storage space and increases the catching capabilities in event tracking. Our experiments conducted on two Datasets show that DTED outperforms the state-of-the-art clustering algorithms and the single-pass algorithms in F1-measure; meanwhile, experiment on standard TDT4 dataset shows promising result.

References

1. HongXiang, D., Ge, X., Jian, X.: An Improved New Event Detection Model. CCIS, pp. 431–437 (2010)
2. Canhui, W., Min, Z., Shaoping, M.: Automatic Online News Issue Construction in Web Environment. In: Proceedings of the 17th International Conference on World Wide Web, pp. 457–466. ACM Press, New York (2008)
3. Yinming, Y., Tom, P., Jaime, C.: A Study of Retrospective and On-line Event Detection. In: Proceedings of the 21th Annual International ACM SIGIR Conference, pp. 28–36. ACM Press, New York (1998)
4. Allan, J., Papka, R., Lavrenko, V.: On-line New Event Detection and Tracking. In: Proceedings of the 21st Annual International ACM SIGIR Conference, pp. 37–45. ACM Press, New York (1998)
5. Qi, H., Kuiyu, C., Lim, E.-P.: Analyzing Feature Trajectories for Event Detection. In: Proceedings of the 30th Annual International ACM SIGIR Conference, pp. 207–214. ACM Press, New York (2007)
6. Van Rijsbergen, C.J.: Information Retrieval, 2nd edn. Buttersworth, London (1979)
7. Steinbach, M., Karypis, G., Kumar, V.: A Comparison of Document Clustering Techniques. In: KDD Workshop on Text Mining (2000)
8. Allan, J., Feng, A., Bolivar, A.: Flexible Intrinsic Evaluation of Hierarchical Clustering for TDT. In: CIKM, pp. 263–270 (2003)
9. Kumaran, G., Allan, J.: Text classification and named entities for new event detection. In: ACM SIGIR Conference (2004)
10. Kuo, Z., Juanzi, L., Gang, W.: New Event Detection Based on Indexing-tree and Named Entity. In: SIGIR, pp. 215–222 (2007)
11. Brants, T., Francine, C., Ayman, F.: A System for New Event Detection. In: Proceedings of the 26th Annual International ACM SIGIR Conference (2003)

Effect of Dimensionality Reduction on Different Distance Measures in Document Clustering

Mari-Sanna Paukkeri[1], Ilkka Kivimäki[2], Santosh Tirunagari[1],
Erkki Oja[1], and Timo Honkela[1]

[1] Department of Information and Computer Science, Aalto University School of Science,
P.O. Box 15400, FI-00076 Aalto, Finland
{mari-sanna.paukkeri,erkki.oja,timo.honkela}@tkk.fi,
santosh.tirunagari@aalto.fi
[2] ISYS/LSM, Université de Louvain
Place des Doyens 1, B-1348 Louvain-la-Neuve, Belgium
ilkka.kivimaki@uclouvain.be

Abstract. In document clustering, semantically similar documents are grouped together. The dimensionality of document collections is often very large, thousands or tens of thousands of terms. Thus, it is common to reduce the original dimensionality before clustering for computational reasons. Cosine distance is widely seen as the best choice for measuring the distances between documents in k-means clustering. In this paper, we experiment three dimensionality reduction methods with a selection of distance measures and show that after dimensionality reduction into small target dimensionalities, such as 10 or below, the superiority of cosine measure does not hold anymore. Also, for small dimensionalities, PCA dimensionality reduction method performs better than SVD. We also show how l_2 normalization affects different distance measures. The experiments are run for three document sets in English and one in Hindi.

Keywords: document clustering, dimensionality reduction, distance measure.

1 Introduction

Document clustering is a basic method in text mining and has been used, for example, for information retrieval [6,11,13] and in explorative research [4]. Document clustering differs from the clustering of other kinds of data in some ways. The textual data needs preprocessing before feature extraction, which influences heavily the resulting term-document matrix. The term-document matrix is very sparse: most of the terms occur in a small subset of the document collection only.

A traditional straightforward way to extract the meaningful features from a document is to calculate the frequencies of occurrence of certain (or possibly all) words in the document. This approach is usually called the bag-of-words model. The frequency distribution is very skewed and follows a power law. The distribution has a small number of words with very high frequency and many words with very low frequency. The most frequent words in each language, such as *the*, *and*, *he* in English, do not usually carry the semantic content of the document. By collecting these terms to a stop word

list and removing them from the data set, the dimensionality of the data matrix can be reduced and the skewness diminished. Nevertheless, stop word removal adds data sparsity, since the most filled feature vectors are removed from the data matrix. Word stemming is another common preprocessing step. Stemming removes inflected forms of a word by cutting the ends of the words. Also words that occur only a couple of times in the data set can be removed, since there is a large amount of them and they do not help much in the clustering. Stemming and low frequency word removal reduce sparsity through combining the features (former) or removal of very sparse feature vectors (latter).

Our contribution in this paper is to show that dimensionality reduction affects the performance of distance measures, especially in small target dimensionalities, and that the superiority of the cosine distance does not hold everywhere. We also show that PCA performs better than SVD in the document clustering task in small dimensionalities and that Bray-Curtis, which is not a very well-known distance measure, performs usually best with PCA in small dimensionalities. We perform experiments with a selection of dimensionality reduction methods and normalization of document data. We also apply the same experiments for two very different languages, English and Hindi.

2   Methods

In this section we discuss dimensionality reduction, different distance measures, and document clustering with k-means algorithm.

2.1   Dimensionality Reduction

Dimensionality reduction is transformation of data into a more compact, low-dimensional representation. It is used to decrease the computational load of clustering and it can also remove redundancy in the data. However, too extensive reduction may result in the loss of information that would be beneficial for the clustering. Three linear techniques are popular for dimensionality reduction: *Principal Component Analysis*, *Singular Value Decomposition*, and a naive approach that selects terms that have largest weight in the document set. We denote by $\mathbf{X}$ the data matrix which contains n-dimensional document vectors $\mathbf{x}_1, \ldots, \mathbf{x}_N$ as its columns. The linear reduction into the target dimensionality D can be achieved by a transformation

$$\mathbf{X}_D = \mathbf{W}_D^\mathsf{T}\mathbf{X}, \tag{1}$$

where the matrix $\mathbf{W}_D$ is of size $n \times D$.

In *Principal Component Analysis* (PCA), the columns of $\mathbf{W}_D$ are chosen as the D eigenvectors corresponding to the D largest eigenvalues of the data covariance matrix. In *Singular Value Decomposition* (SVD) of the data matrix $\mathbf{X} = \mathbf{V}\mathbf{\Sigma}\mathbf{U}^\mathsf{T}$, the matrix $\mathbf{W}_D$ is formed by choosing the D column vectors of the matrix $\mathbf{V}$ corresponding to the largest singular values of $\mathbf{X}$. It is worth noting that if the data vectors are centered, i.e., $\mathrm{E}(\mathbf{X}) = 0$, PCA and SVD produce the same projection, but otherwise the methods are not equivalent. PCA is not well suited for text data, because computation of the

covariance matrix involves centering the data matrix, which fills the sparse data. Anyway, we wanted to experiment with PCA because of its popularity. The use of SVD for term-document matrices is better known as *Latent Semantic Analysis* (LSA) [2] in the IR community.

As the naive approach, the dimensionality was reduced by selecting the terms with the highest *term frequency – inverse document frequency* (tf-idf) weight sums over a document set. The kth tf-idf weight is $x_{k,i} = f_{k,i} \cdot \log(N/N_k)$, where $f_{k,i}$ is the frequency of the term t_k in the document i, N is the total number of documents in the collection and N_k is the number of documents where the term t_k appears. Considering the Equation (1), dimensionality reduction with tf-idf weight sums is choosing $\mathbf{W}_D$ so that each column vector has one of the elements corresponding to the D terms with the highest tf-idf scores equal to 1 and the rest of the elements are zeros.

2.2 Distance Measures

We interpret the term *distance measure* in a space Ω as a function $d : \Omega \times \Omega \rightarrow \mathbb{R}$ that is non-negative, symmetric, and reflexive. Some of the distance measures used in this study satisfy also stricter conditions, e.g., the properties of a metric. However, those conditions are not always necessary for practical purposes, like clustering.

A number of distance measures that will be studied in the document clustering task are listed in Table 1. Some of them have been tested earlier for document clustering [5,10,15,17]. The distance measures are defined for two data vectors

Table 1. Distance measures between column vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ of the data matrix $\mathbf{X}$

Measure	Distance d_{ij}	Measure	Distance d_{ij}
Euclidean	$\sqrt{(\mathbf{x}_i - \mathbf{x}_j)^{\mathsf{T}}(\mathbf{x}_i - \mathbf{x}_j)}$	Spearman[3]	$1 - \dfrac{(\mathbf{r}_i - \bar{\mathbf{r}})^{\mathsf{T}}(\mathbf{r}_j - \bar{\mathbf{r}})}{\|\mathbf{r}_i - \bar{\mathbf{r}}\|_2 \|\mathbf{r}_j - \bar{\mathbf{r}}\|_2}$
Standardized Euclidean[1]	$\sqrt{(\mathbf{x}_i - \mathbf{x}_j)^{\mathsf{T}}\mathbf{V}^{-1}(\mathbf{x}_i - \mathbf{x}_j)}$	Bray-Curtis	$\dfrac{\sum_{k=1}^n |x_{k,i} - x_{k,j}|}{\sum_{k=1}^n (x_{k,i} + x_{k,j})}$
City block	$\sum\limits_{k=1}^n |x_{k,i} - x_{k,j}|$	Bray-Curtis 2	$\dfrac{\sum_{k=1}^n |x_{k,i} - x_{k,j}|}{\sum_{k=1}^n (|x_{k,i}| + |x_{k,j}|)}$
Chebychev	$\max\limits_{1 \le k \le n} \{|x_{k,i} - x_{k,j}|\}$		
Cosine	$1 - \dfrac{\mathbf{x}_i^{\mathsf{T}} \mathbf{x}_j}{\|\mathbf{x}_i\|_2 \|\mathbf{x}_j\|_2}$	Canberra[4]	$\sum\limits_k \dfrac{|x_{k,i} - x_{k,j}|}{|x_{k,i}| + |x_{k,j}|}$
Correlation[2]	$1 - \dfrac{(\mathbf{x}_i - \bar{\mathbf{x}}_i)^{\mathsf{T}}(\mathbf{x}_j - \bar{\mathbf{x}}_j)}{\|\mathbf{x}_i - \bar{\mathbf{x}}_i\|_2 \|\mathbf{x}_j - \bar{\mathbf{x}}_j\|_2}$		

[1] $\mathbf{V}$ is a $n \times n$ diagonal matrix of variance of the kth variable on its kth diagonal element
[2] $\bar{\mathbf{x}}_i$ is the mean vector of elements $\mathbf{x}_i$
[3] $\mathbf{r}_i$ is the coordinate-wise rank vector of $\mathbf{x}_i$ and $\bar{\mathbf{r}}$ contains mean ranks of an n-dimensional vector, i.e. $(n+1)/2$
[4] The sum is taken over those k, for which $|x_{k,i}| + |x_{k,j}| \neq 0$

$\mathbf{x}_i = [x_{1,i}, \ldots x_{n,i}]^{\mathsf{T}}$ and $\mathbf{x}_j = [x_{1,j}, \ldots x_{n,j}]^{\mathsf{T}}$. City block, Euclidean, and Chebychev distances are the standard special cases of the l_p-metric [3]. The standardized Euclidean

distance differs from the Euclidean distance by having the variance of each variable normalized into unity. The cosine distance measures the angle between two data vectors. Correlation distance works in a similar way, but it is more commonly used to compare variables of a space, rather than data vectors in the space, like in this work. Correlation distance is based on the formulation of Pearson's correlation coefficient between two variables. Spearman distance is based on Spearman's rank correlation coefficient. We also try out Bray-Curtis and Canberra distances, that are less conventional for clustering document collections. They originate from ecological and environmental research [1]. The Bray-Curtis distance is for non-negative data but the dimensionality reduction with SVD and PCA results in data with negative values. Thus we also use a modified version of Bray-Curtis (referred to Bray-Curtis 2) which uses the absolute values of the vector components in the denominator.

We first illustrate with artificial data how dimensionality reduction into different target dimensionalities affects the behavior of the distance measures: we observe how the neighborhoods of data vectors are preserved during the dimensionality reduction. This relates to the notion of distance preservation, which is often chosen as a criterion for nonlinear dimensionality reduction methods [8]. Our artificial data resembles real term-document data to some extent: a sparse data matrix $\mathbf{X}_{\text{toy}} \in \mathbf{R}^{1000 \times 100}$ contains 90% of zero elements and 10% of random numbers distributed uniformly between $[0, 1]$. The vectors are normalized to unity in l_2 norm. The nearest neighbor of each data vector is calculated for each distance measure separately. After dimensionality reduction with SVD, PCA, and tf-idf, the nearest neighbors are searched again. The mean ratios of the preserved neighbors, calculated over 20 repetitions of the experiment, are shown for each dimensionality reduction method and distance measure in Figure 1. The maximum target dimensionality is 100, the number of documents, for both SVD and PCA since the rank of the data matrix is 100. The dimensionality reduction with the tf-idf weights is plotted up to 1000, the original dimensionality. Figures of SVD and PCA are so similar that we show just the former.
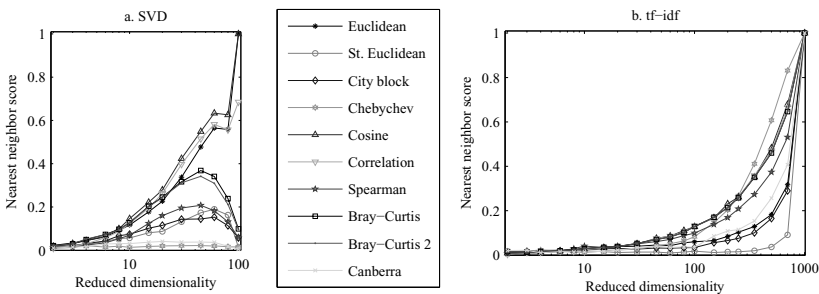
Fig. 1. Mean Nearest neighbor score of toy data with a. SVD and b. tf-idf

In target dimension 100, SVD and PCA correspond to a rotation of the coordinate axes and thus Euclidean and cosine distances preserve all the neighbors. Correlation modifies the vectors before comparing the angles between the vectors and gets a slightly lower score than cosine in dimension 100. These distance measures perform best in the

neighbor-preservation task with SVD and PCA dimensionality reduction for all target dimensionalities. The other distance measures are sensitive to rotation of the coordinates and thus give low nearest neighbor scores at dimension 100. With low target dimensionalities up to dimensionality 50, the Bray-Curtis measures give almost as good results as Euclidean, cosine, and correlation distances.

Tf-idf is not able to preserve the neighborhood of data vectors below target dimensionality 100: less than 15% of the neighbors are preserved with all distance measures. For dimensionalities between 100 and 1000 the best scores are obtained with Chebychev, cosine, correlation and both the Bray-Curtis measures, respectively.

2.3   K-Means Clustering

Different clustering methods have been analyzed and compared in the literature of the document clustering task: e.g., hierarchical partitional and hierarchical agglomerative clustering [20], one-dimensional Self-Organizing Map and graph partitioning [17], and more generally spectral clustering (see tutorial in [9]). Latent Semantic Indexing (LSI) [2] and spectral clustering transform the data into a new space and after that, use e.g. k-means clustering. K-means is a popular method in document clustering [15,5]. In our analysis of multiple dimensionality reduction methods and several distance measures, we use k-means due to the simplicity, efficiency and fairly good performance in general. See a comprehensive analysis of document clustering methods in [11].

The k-means algorithm is described e.g., in [7]. Given a set of data vectors $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \subset \mathbb{R}^D$ and the number of clusters k, the algorithm tries to find a set of k vectors $\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_k\} \subset \mathbb{R}^D$ and a partition of the data set $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$ such that the cost function

$$\mu(\mathcal{C}) = \sum_{j=1}^{k} \sum_{\mathbf{x}_i \in C_j} d(\mathbf{x}_i, \mathbf{c}_j)^2 \tag{2}$$

is minimized with respect to a given distance measure d.

The assumption with k-means when using the Euclidean distance is that the data contains spherical clusters. With the other distance measures the clusters are assumed to be of different shape, e.g. standardized Euclidean finds elliptical and city block cubical clusters. These assumptions, however, do not often hold for natural data sets. K-means clustering can be performed efficiently by iteratively updating each cluster centroid c_i as the mean vector of the vectors belonging to the cluster C_i in the Euclidean case. This greedy algorithm usually results in fast convergence, although it may easily stop at a local minimum of the cost function, depending on the cluster centroid initialization [18]. In addition to Euclidean distance, the cluster centroid is defined as the mean vector for Standardized Euclidean and Chebychev distances. For the other distance measures, the update of the cluster centroids must be modified. For cosine, correlation, and Spearman distances the mean vector is normalized. For the city block, Canberra, and Bray-Curtis measures the cluster centroids are updated as the component-wise median vector of the vectors in a cluster. If kernel k-means were utilized in this task the kernel could have been selected so that the distances between data items were replaced with vector similarities and thus updating the cluster centroids would have been easy.

3   Experiments

We have conducted our experiments for three data sets in English and one set in Hindi. After preprocessing the data, a set of dimensionality reduction methods are applied. The clustering is run with the k-means algorithm using a number of distance measures The number of clusters k is fixed according to the number of categories present in a document collection. For evaluation we use the *Normalized Mutual Information* (NMI) score (used also in [16] and a variant in [19]). All the results are obtained by calculating the mean NMI scores after running the clustering 20 times with a given distance measure. The centroids are initialized by selecting randomly k vectors from the data set and the resulting clusters are evaluated with the class labels.

3.1   Data and Preprocessing

First of our three English data sets is REUTERS R8[1] with 8 categories, collected originally from Reuters newswire. The categories are very skewed, varying from 41 documents in the smallest category to 2840 documents in the largest category. The second data set is from the same web page, 20 Newsgroups (called NEWSGROUP), containing about 11 000 newsgroup articles in 20 categories. Each category has almost the same number of documents. The third document collection is CLASSIC4[2] set of documents from four sources. As our HINDI data set we collected online news articles from a Hindi news site 24duniya[3]. The articles are from 4 categories: Automobile, Business, Health and Election and they have been collected manually between 4–7 February 2011.

The text in each data set was lowercased and punctuation was removed. Next, a list of stop words were removed. Our stop word lists contain 422 words in English (including single letters) and 97 words in Hindi. The remaining words were stemmed using a Java implementation[4] of the Porter's stemmer [12] for English and a simple stemmer for Hindi, following the approach described in [14]. From the remaining data, words that occur 5 times or less in the data set were removed. Finally, the words were weighted with the tf-idf weighting scheme and the weights normalized with l_2 norm. The sizes of the data sets after preprocessing are shown in Table 2. The dimensionality (number of features) and average length of documents are calculated after preprocessing. REUTERS and CLASSIC documents are fairly short compared to the other sets.

Table 2. The characteristics of the used data sets

Data set	Documents	Dimensionality	Categories	Avg. length
NEWSGROUP	11293	16555	20	126.2
REUTERS	5485	4199	8	58.4
CLASSIC	4774	4130	4	57.2
HINDI	768	2669	4	123.2

---

[1] http://web.ist.utl.pt/~acardoso/datasets/
[2] ftp://ftp.cs.cornell.edu/pub/smart/
[3] http://www.24dunia.com/hindi.html
[4] http://tartarus.org/~martin/PorterStemmer/

Table 3. Mean NMI scores of clustering without dimensionality reduction for unnormalized (U) and l_2-normalized (N) data. The best result for each data set is shown in bold face.

Distance measure	NEWSGROUP		REUTERS		CLASSIC		HINDI	
	U	N	U	N	U	N	U	N
Euclidean	0.063	0.512	0.209	0.444	0.240	0.548	0.098	0.403
Standardized Euclidean	0.040	0.098	0.110	0.092	0.074	0.065	0.049	0.095
City block	0.055	0.072	0.180	0.231	0.047	0.069	0.105	0.199
Chebychev	0.162	0.133	0.117	0.114	0.172	0.156	0.057	0.067
Cosine	**0.607**	**0.607**	0.512	0.512	**0.627**	**0.627**	0.422	0.422
Correlation	0.604	0.604	0.513	0.513	0.509	0.509	**0.434**	**0.434**
Spearman	0.429	0.429	**0.530**	**0.530**	0.585	0.585	0.410	0.410
Bray-Curtis	0.144	0.156	0.261	0.258	0.117	0.128	0.151	0.256
Bray-Curtis 2	0.144	0.156	0.261	0.258	0.117	0.128	0.151	0.256
Canberra	0.050	0.044	0.202	0.194	0.050	0.044	0.029	0.024

3.2 Results

The clustering results of original data sets without dimensionality reduction are shown in Table 3. Cosine, correlation and spearman measures perform best in the classification task for all data sets. With l_2-normalization, also Euclidean gives good results.

In Figure 2, the effect of l_2-normalization is shown with PCA dimensionality reduction on the REUTERS data and SVD on the NEWSGROUP data. The dimensionality was reduced into target dimensionalities ranging between 2 and 1000 and the clustering run with k-means using all the distance measures. Normalization affects neither cosine nor
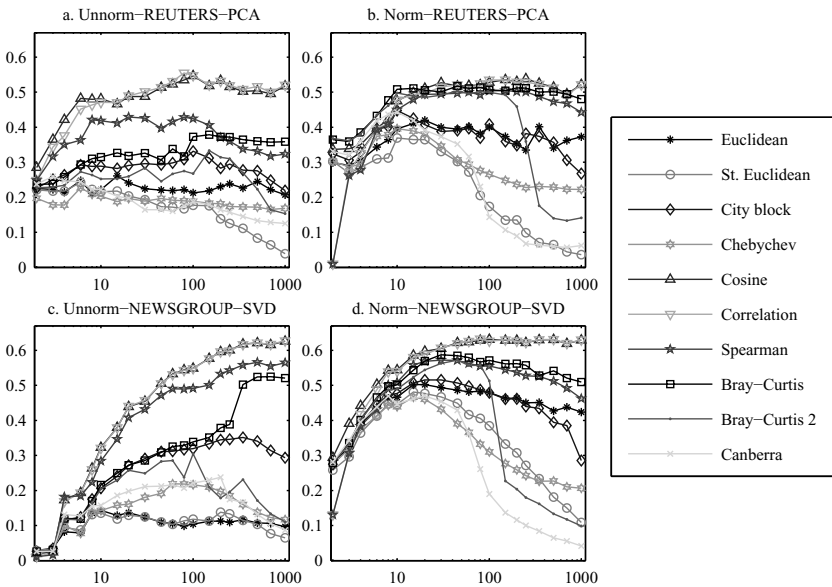
Fig. 2. Mean NMI of unnormalized and l_2-normalized data for target dimensions 2–1000
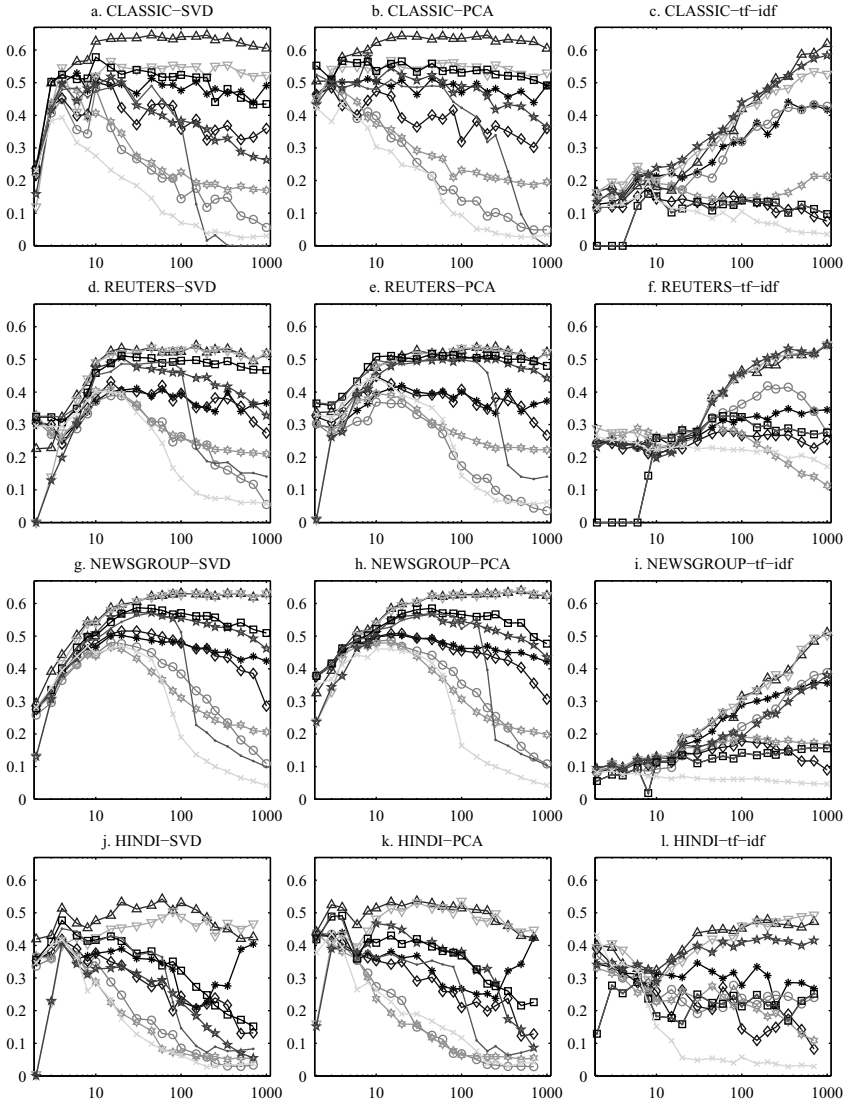
Fig. 3. Mean NMI after dimensionality reduction into dimensions 2–1000 for normalized data. See Fig. 2 for the legend.

correlation measures in large dimensionalities but decreases the performance below dimensionality 10 with PCA dimensionality reduction. On the other hand, all the other dimensionality reduction methods perform better with normalized data for almost all dimensionalities. Especially Bray-Curtis, both the Euclidean measures, and Spearman benefit from the normalization. An exception is Canberra which reaches lower scores with normalized than unnormalized data above dimensionality 100.

Since normalization generally helps document clustering, the rest of the results are given for normalized data. The mean NMI results after SVD, PCA and tf-idf dimensionality reduction are shown in Figure 3 for all four data sets: CLASSIC, REUTERS, NEWSGROUP, and HINDI. As predicted already with the toy data, cosine and correlation distances perform best almost always in target dimensionalities above 10. For target dimension 10 or below, all the distance measures give very similar results and the dimensionality reduction method has the major role in the performance. For the small dimensionalities, PCA performs better than SVD, which may be due the to fact that PCA minimizes the mean-square error between vectors but SVD minimizes the Frobenius distance between matrices. For dimensionalities above 10 the differences between SVD and PCA are very small, except for Spearman which works better with PCA than SVD. The performance of Bray-Curtis is very stationary and usually within the three best measures throughout the target dimensionalities with SVD and PCA. Spearman performs well with tf-idf dimensionality reduction and pretty well also with SVD and PCA. Also standardized Euclidean works with tf-idf but gives poor results with other dimensionality reduction methods. Canberra, Bray-Curtis 2, and Chebychev are always among the worst-performing measures. The performance of Bray-Curtis works best with normalized data and PCA or SVD dimensionality reduction. Standardized Euclidean, City block, Chebychev, Bray-Curtis and Canberra did very poor performance in the original dimensionality (Table 3) but did benefit from SVD and PCA dimensionality reduction.

4   Conclusions

In this work, we studied the effect of dimensionality reduction on different distance measures in document clustering. As a result, we found that for document vectors of very small dimensionality, such as 10 or below, the commonly used cosine distance with SVD dimensionality reduction does not always give the best results. For example, Bray-Curtis distance with PCA dimensionality reduction should be used instead. Generally, for small target dimensionalities, PCA performs better than SVD for l_2-normalized data, independently of the distance measure. With dimensionality reduction according to tf-idf weights, the Spearman distance gave very good results for all dimensionalities. As an overall result, we found that cosine and correlation distance measures give us good results with all three dimensionality reduction methods: SVD, PCA, and tf-idf. Bray-Curtis and Spearman are usually the next ones. Euclidean and Standardized Euclidean produced inconsistent results depending on normalization and the applied dimensionality reduction method. Canberra, the modified Bray-Curtis, and Chebychev were among the worst-performing measures.

The dimensionality reduction method and the target dimension has a substantial effect on the clustering performance. In this work, the behavior of different dimensionality reduction methods and distance measures were quite consistent over different data sets. In the future, this work can be carried out for a larger variety of different languages and domains. The results reported in this paper can also be utilized in other studies that use document clustering as part of the application.

References

1. Clarke, K.R., Somerfield, P.J., Chapman, M.G.: On resemblance measures for ecological studies, including taxonomic dissimilarities and a zero-adjusted Bray-Curtis coefficient for denuded assemblages. Experimental Marine Biology and Ecology 330(1), 55–80 (2006)
2. Deerwester, S.C., Dumais, S.T., Landauer, T.K., Furnas, G.W., Harshman, R.A.: Indexing by Latent Semantic Analysis. Journal of the American Society of Information Science 41(6), 391–407 (1990)
3. Deza, M.M., Deza, E.: Encyclopedia of distances. Springer, Heidelberg (2009)
4. Honkela, T., Kaski, S., Lagus, K., Kohonen, T.: WEBSOM – Self-organizing maps of document collections. In: Proceedings of WSOM 1997, pp. 310–315 (1997)
5. Huang, A.: Similarity measures for text document clustering. In: Proceedings of NZCSRSC 2008, pp. 49–56 (2008)
6. Jardine, N., van Rijsbergen, C.: The use of hierarchic clustering in information retrieval. Information Storage and Retrieval 7(5), 217–240 (1971)
7. Kanungo, T., Mount, D.M., Netanyahu, N.S., Piatko, C.D., Silverman, R., Wu, A.Y.: An efficient k-means clustering algorithm: Analysis and implementation. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7), 881–892 (2002)
8. Lee, J.A., Verleysen, M.: Nonlinear Dimensionality Reduction. Springer, Heidelberg (2007)
9. Luxburg, U.: A tutorial on spectral clustering. Statistics and Computing 17(4), 395–416 (2007)
10. Madylova, A., Öğüdücü, Ş.G.: Comparison of similarity measures for clustering Turkish documents. Intelligent Data Analysis 13(5), 815–832 (2009)
11. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press (2008)
12. Porter, M.: An algorithm for suffix stripping. Program 14(3), 130–137 (1980)
13. Radovanović, M., Nanopoulos, A., Ivanović, M.: On the existence of obstinate results in vector space models. In: Proceedings of SIGIR 2010, pp. 186–193. ACM (2010)
14. Ramanathan, A., Rao, D.D.: A lightweight stemmer for Hindi. In: Proceedings of EACL 2003. ACL (2003)
15. Schenker, A., Last, M., Bunke, H., Kandel, A.: Comparison of Distance Measures for Graph-Based Clustering of Documents. In: Hancock, E.R., Vento, M. (eds.) GbRPR 2003. LNCS, vol. 2726, pp. 202–213. Springer, Heidelberg (2003)
16. Strehl, A., Ghosh, J.: Cluster ensembles — a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research 3, 583–617 (2003)
17. Strehl, A., Ghosh, J., Mooney, R.: Impact of similarity measures on web-page clustering. In: Proceedings of AAAI 2000, pp. 58–64 (2000)
18. Theodoridis, S., Koutroumbas, K.: Pattern Recognition, 4th edn. Academic Press (2008)
19. Xu, W., Liu, X., Gong, Y.: Document clustering based on non-negative matrix factorization. In: Proceedings of SIGIR 2003, pp. 267–273. ACM (2003)
20. Zhao, Y., Karypis, G.: Hierarchical clustering algorithms for document datasets. Data Mining and Knowledge Discovery 10(2), 141–168 (2005)

Diversifying Question Recommendations in Community-Based Question Answering

Yaoyun Zhang, Xiaolong Wang, Xuan Wang, Ruifeng Xu, and Buzhou Tang

Key Laboratory of Network-oriented Intelligent Computation
Harbin Institute of Technology Shenzhen Graduate School
Shenzhen 518055, China
{wangxl,wangxuan}@insun.hit.edu.cn,
{xiaoni5122,xuruifeng.hitsz,tangbuzhou}@gmail.com

Abstract. Question retrieval is an important research topic in community-based question answering (QA). Conventionally, questions semantically equivalent to the query question are considered as top ranks. However, traditional question retrieval technique has the difficulty to process the users' information needs which are implicitly embedded in the question. This paper proposes a novel method of question recommendation by considering user's diverse information needs. By estimating information need compactness in the question retrieval results, we further identify the retrieval results need to be diversified. For these results, the scores of information retrieval model, the importance and novelty of both question types and the informational aspects of question content, are combined to do diverse question recommendation. Comparative experiments on a large scale real community-based QA dataset show that the proposed method effectively improves information need coverage and diversity through relevant questions recommendation.

Keywords: Question answering, Community-based question answering, Question recommendation, Question diversity, Information need.

1 Introduction

Recent years see a booming of online question answering (QA) community portals, such as yahoo answers! and Baidu Zhidao. Such portals accumulate huge question and accepted answer archives, from where users can directly search and find similar questions with accepted answers.

A lot of research is conducted on question retrieval [1] to enhance the precision of retrieving questions semantically equivalent to the posted question. The retrieved questions tend to be homogenous and express the same information needs (*infoNeeds*) as the posted question.

However, diverse *infoNeeds* are potentially embedded in the question. On the one hand, there is a gap exists between the *infoNeeds* expressed in the question and in the user's consciousness [2]. The user may be unable to express his/her *infoNeeds* completely because s/he is in lack of domain knowledge, or the system function is

limited. On the other hand, users with different intentions may have different *infoNeeds* for the same question [3].

q1: *Which* bank *provides the best credit card*?

For the example question q1, if the user only wants to confirm the bank he knows, the name of the bank is enough for an answer. While the user plans to open a credit card, he may want to see detailed description and comparison between credit card services of different banks, and thus the answers for following question q2 and q3 should be useful. Nevertheless, such *infoNeeds* are not given explicitly in the question text of q1.

q2: Please *give me a comparison between the credit card service of Citibank or HSBC*?

q3: How *to choose a credit care*?

In QA systems, question type is commonly used as the general representation of user *infoNeeds* [4], while the question content sets the context (by question topic) and specifies the *infoNeed* (by question focus). As can be seen, q2 and q3 are different from q1 both in question type (q1: entity type; q2: comparison type; q3: method type) and content. These questions may not be represented to the users by the conventional question retrieval systems, because they are not semantically equivalent to q1. Therefore, only returning questions semantically homogenous with the posted question is not sufficient to fulfill the potential diverse *infoNeeds* of users. The coverage of *infoNeeds* in the retrieved questions should be enhanced.

To fulfill the implicit and diverse *infoNeeds* of the question issued by average users, this paper propose an novel question recommendation algorithm, with diversification *infoNeeds* both in question types and content. By estimating information need compactness in the question retrieval results, we identify the retrieval results need to be diversified. For the question in these results, this proposed algorithm first recognizes its question type, topic and focus. The diverse *infoNeeds* are then modeled in the question type and content separately by probability models. Those factors are then combined linearly in the diversification algorithm for question recommendation. By returning recommended questions and corresponding accepted answers to users, this algorithm can provide more relevant and diverse information.

Previous works on information diversity [5] mainly target on short queries in the information retrieval scenario. Their focuses are to mine the different interpretations of ambiguous queries or subtopics for a broad-sense query. Achananuparp et al. attempted to diversify the aspects of the answer to complex questions [6], while their focus is still on the short information nuggets returned by search engines. [7] recommended questions which have similar topics and different focus with the query question. However, they didn't consider the diversity or coverage of *infoNeeds* in the recommendations. As far as we know, this work is the first attempt to diversify question recommendation. Extensive comparative experiments conducted on a large scale dataset collected from real community-based QA portals to validate the effectiveness of the proposed method.

The rest of this paper is organized as follows. Related work is introduced in Section 2; Section 3 explains the proposed algorithm in detail; Section 4 describes experimental settings and comparative results; the conclusion is given in Section 5.

2   Diversification Algorithm of Question Recommendation

2.1   Problem Definition

For a query question *qy*, its *infoNeeds* are split into two parts: (a) the types of *infoNeeds*, represented by question types; (b) the relevant informational aspects expressed in the question content.
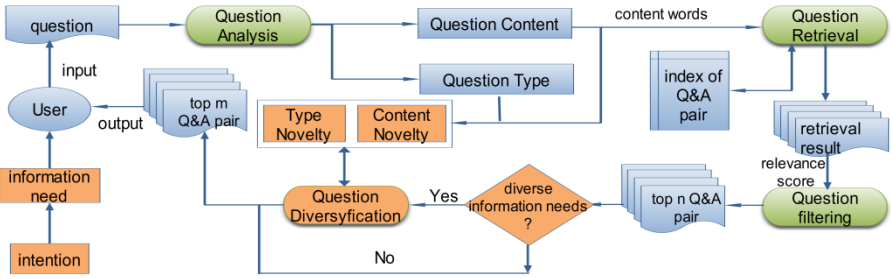
Fig. 1. Procedure of diversifying question recommendations

The procedure of question recommendation diversification algorithm is given in Fig.1. Suppose we have a collection of pairs of question q_i with accepted answers a_i, denoted as $C=\{ (q_i ,a_i)\}$. q_i is treated as a document and an index is built for questions in *C*. *R* is the set of top *n* result questions by searching *qy* in the index using an information retrieval (*IR*) model.

As explained in the introduction section, before trying to recommend diverse questions to the user, we first estimates whether *qy* has diverse *infoNeeds*, by computing *infoNCompact(qy)*, i.e., the information need compactness of *qy*.

If *qy* is considered to have diverse *infoNeeds* in the above step, our aim is to find an ordered list of *m* (*n≥m*) questions from *R*, denoted as *relist*. Questions in *reList* are ranked by *rel_div_score(q)*, which takes into account the following factors:

1. ***relevance(q,qy)***: the relevance score between *q* and *qy* derived by the employed *IR* model; 2. ***importance (type(q)|qy)***: the importance of the information need represented by the question type of *q* is among all the potential information needs of *qy*; 3. ***novelty(type(q)|qy)***: the novelty of the question type of q against the questions already ranked in *reList*; 4. ***importance(content(q)|qy)***: the importance of the information in the question content of *q* is among all the information relevant to *qy*; 5. ***novelty(content(q)|qy)***: the novelty of the question content of *q* against the questions already ranked in *reList* .

2.2   Question Recommendation Procedure

Information Need Compactness

Intuitively, if the *infoNeed* of the query question is homogenous, the retrieved questions will be similar to each other. We use Intra-Cluster Similarity to measure the *infoNeed* compactness of the query and result questions returned by an information

retrieval model. First, we consider a question as a document and convert questions into vectors in the vector space model: di=(x_{i1}, x_{i2}, ...). Each component of the vectors represents a distinct unigram and is weighted by tf.idf of this unigram. For each group of query and result questions (denoted as D_{qy}), the *infoNeed* compactness is formulated as:

$$infoNCompact(qy) = \frac{1}{|D_{qy}|}\sum_{d_i \in D_{qy}} \cos(d_i, o), \text{ where the centroid } o = \frac{1}{|D_{qy}|}\sum_{d_i \in D_{qy}} d_i \tag{1}$$

Diverse questions are recommended for *qy* with *infoNCompact*(*qy*) no larger than Φ.

Question Type Classification

To ensure that the recommended questions have diverse question types, a question type prediction model is built using supervised machine learning methods with probabilistic output *p(type$_i$|qy)*.

Since real questions are a mixture of factoid as well as complex ones, 14 types are defined in this paper, which are *location, person, time, quantity, thing, alternative, definition, comparison, description, procedure, reason, yesNo, abstractEnity* and *other*. The first 5 types are factoid, and the latter 8 are complex. Detailed feature descriptions are referenced to [8]. Thus we have

$$importance(type(q)|qy) = p(type(q)|qy) \tag{2}$$

If *importance*(*type*(*q*)|*qy*) is larger than a threshold θ (θ is empirically set to 0.5), then the type is considered as a relevant type of *infoNeeds* of *qy*.

Informational Aspects Mining in Question Content

The content words (namely nouns, verbs and adjectives) are extracted from *qy* as the question content. The question content consists of two different information roles: question topic which presents the major context/constraint of a question and characterizes users' interest; and question focus which presents certain aspect (or descriptive features) of the question topic [8]. Hence, by diversifying the question content, we aim to diversify the informational aspects of the question topic.

Question Topic and Focus Determination

We first compute the local tf.idf (taking *qy* and the top n search result as the document set, labeled as *loc_tf.idf*) and global tf.idf (taking the whole indexed questions as the document set, labeled as *glob_tf.idf*) of the content word (labeled as *cw*) in *qy*.

$$loc\_tf.idf = tf_{cwi} * \log(loc\_docNum)) / loc\_df_{cwi} \tag{3}$$

$$glob\_tf.idf = tf_{cwi} * log(glob\_docNum)) / glob\_df_{cwi} \tag{4}$$

Next, the rank of content words *cw* in *qy* is determined by:

$$cwRank = rank(glob\_tf.idf) / rank(loc\_tf.idf) \tag{5}$$

where rank(*loc_tf.idf*) and rank(*glob_tf.idf*) are the ranked position of *cw* according to local and global tf.idf value, respectively.

We consider the top 50% content words (with additional constraint of no more than 3 words) ranked by *cwRank* as the topic words, and the left ones are focus words.

Informational Aspects of Question Topic

Next, the informational aspects of question topic of *qy* are extracted. For each q_i in the top 100 search result questions, which contains at least one topic word (*tw*) of *qy*, the relevance of cw_i in q_i to *qy* is computed as:

$$relevance(cw_i, qy) = \sum_j PMI(cw_i, tw_j) / |tw_j| \qquad (6)$$

where *PMI* denotes the pointwise mutual information [9] between two words.

The top 50 cw_i ranked by *relevance*(cw_i, *qy*) are considered as the potential informational aspects of question topic.

However, the aspects extracted may have synonyms, or express similar meanings. Therefore, the Star Clustering [10] is applied to merge homogeneous topic aspects. The key factor in Star Clustering is the similarity criterion based on which two vertices (i.e., informational aspects) are connected as an edge. This paper defines the criterion as *ContextSimi*(cw_i; cw_j) > *a* (*a* is empirically set to 0.05). The word before and after cw_i are considered as its context. *ContextSimi*(cw_i; cw_j) measures the context similarity between cw_i and cw_j, which is computed follows the algorithm in [11].

After the clustering process, the cluster centroids are considered as the informational aspects of topic. Among those words, *relevance(cw_i, qy)* are normalized as a probability distribution $p(cw_i, | qy)$, denoted as *importance(cw_i, | qy)*. For a question *q* containing cw_i:

$$importance(content(q) | qy) = Avg \left(\sum_i importance(cw_i | qy)\right) \qquad (7)$$

Diversification Algorithm

Input: top *n* (*n*=100) question & answer pairs retrieved by the *IR* model
Output: top *m* (*m*=10) question & answer pairs with diverse *infoNeeds*
Procedure:

Denote *relist* as the list of questions already re-ranked by diversity, for *q* in the remaining retrieved questions:

(1) Compute type and content novelty of *q*, denoted as *Novelty(T_q)* and *Novelty(C_q)*:

$$Novelty(T_q) = avg\left(\sum importance(type(q) | qy)\frac{1}{e^{df_{type(q)}+1}}\right), \exists\, type(q) \in Type(InfoNeeds\ of\ qy) \qquad (8)$$

$$Novelty(C_q) = avg\left(\sum importance(ck | qy)\frac{tf_{ck}}{e^{df_{ck}+1}}\right), \exists\, ck\ in\ q,\ ck \in aspect(Topic\ of\ qy) \qquad (9)$$

where $df_{type(q)}$ is the frequency of *type(q)* in *reList*, and *type(q)* is one of the relevant *InfoNeed* types of *qy*; df_{ck} is the frequency of the content word *ck* in *reList*, and *ck* is one of the informational aspects of the query topic.

(2) Then *rel_div_score(q)* is computed as :

$$rel\_div\_score(q) = (1 - \lambda_1 - \lambda_2)*relevance(q, qy) + \lambda_1 * Novelty(T_q) + \lambda_2 * Novelty(C_q) \qquad (10)$$

where λ_1 and λ_2 are experiment weights and $0 \le \lambda_1, \lambda_2, \lambda_1 + \lambda_2 \le 1$. The question with the highest *rel_div_score(q)* is added to *reList*.

(3) Repeat the above procedure until the *reList* with top m question&answer pairs are returned to the user.

---

3   Evaluation and Discussions

3.1   Experiment Dataset and Settings

Dataset: Questions with accepted answers are collected from Yahoo! Knowledge and Baidu Zhidao, respectively. After removing invalid questions, more than 1,380,000 postings are obtained. The title of the posting is used as the question.

4800 questions are chosen randomly from the collection to build the question type prediction model. Another 100 questions are chosen randomly from the collection as the query questions.

Experimental Platform: Unigrams of content words of the remaining questions are indexed by Lucene, to build a question retrieval system. Vector space model, the Okapi BM25 model and language model are used as the information retrieval model, respectively. The best performance achieved by the BM25 model is reported here. More state-of-the-art retrieval models, such as [1, 12] will be employed in the future work.

Relevance Set: The relevance set is built with the aim on mind that the recall of relevant questions should be enlarged, so that the coverage of *infoNeeds* will be enlarged. Therefore, the questions that are not semantically equivalent to the query but contain relevant information are also included in the relevance set.

Information Need Annotation: Since the aim of proposed diversification algorithm is to accommodate the potential difference between the *infoNeeds* of different users, three annotators annotate relevant *infoNeeds* individually. A list consisting of the 14 question types and a list consisting of the information aspects collected by the algorithm in Section 2.2 are provided to the annotators, from which they can choose relevant *infoNeeds*.

Thus, three different *infoNeeds* sets of the query questions are generated. The algorithm performance is evaluated on each *infoNeeds* set separately, and the average performance is reported.

Evaluation Criteria: Three evaluation metrics used in the TREC diverse track are employed in this paper: *MAP_IA*, *P@5_IA* [13], and *α-nDCG@10* (α=0.5) [14]. The weights in the evaluation criteria are equal among different *infoNeeds* currently.

Evaluated Algorithms: (1) *Retrieve_M*: the performance of using only the information retrieval model; (2) *MDL-based question recommendation*: [7] recommends questions which have similar topics and different focus with the query question. (3) *MMR*: *MMR* [15] measures the relevance and diversity independently and provides a linear combination, called "marginal relevance", as the metric. (4) *Type_Div*: the proposed diversification algorithm, but only uses the question type for diversity. (5) *Content_Div*: the proposed diversification algorithm, but only uses the question content for diversity. (6) *Type&Content_Div*: the proposed diversification algorithm in this paper.

To achieve a fair comparison, all the diversity parameters in the above algorithms are set to 0.5, i.e., the relevance and diversity of the question are equally weighted. For *Type&Content_Div*, since we have two parameters, λ_1 and λ_2 are set equally to 0.25.

3.2   Experimental Results

Evaluations on Question Analysis

Support Vector Machine (*SVM*) is applied for question classification, using the default parameters of the Rainbow toolkit. The 5-fold cross-validation performance is 91.43%. The question type prediction probability output by *SVM* is employed in this paper.

Totally 790 informational aspects are collected for the 100 query questions, out of which 133 aspects are annotated as irrelevant by at least two annotators. Thus, the precision of the informational aspects are 83.2%. This result indicates that the real questions collected on-line can effectively reflect the *infoNeeds* of average users.

Effectiveness of Information Need Compactness Recognition

Questions with *infoNeed* Compactness larger than $\Phi=0.003$ are considered unnecessary to precede the question recommendation step. 51 questions are left as the *infoNeed*_diverse questions.

We use the relevant *infoNeeds* sets annotated by the three annotators to measure the effectiveness of the algorithm of *infoNeed* compactness. Questions satisfying the following two conditions are considered as *infoNeed*_compact: (1) having only one single relevant question types; (2) having only one or two relevant informational aspects, which are similar to the focus of the query question, or having no relevant informational aspects at all. Table 1 gives the precision and recall of the *infoNeed* compactness algorithm.

Table 1. Comparison between *infoNeed* compactness algorithm and manual annotation

	annotated as compact	annotated as diverse	precision
recognized as compact	26	23	#
recognized as diverse	9	42	82.35%
recall	#	64.66%	

As can be seen, the precision of recognizing questions with diverse *infoNeed* are high enough for practical use. For the 9 precision errors, most of the questions have uncommon topics. The retrieved questions are irrelevant to the query but contain diverse information. The recall are much lower, some questions have large amount of redundancy in the question collection, increasing the number of search results for compactness computation may be a solution.

For the performance of question recommendation diversification, we only report results on the query set recognized by the compactness algorithm as *infoNeed* diverse.

Experimental Comparisons

Table 2 lists the evaluation performance of different algorithms. The performance of only considering the diversity of question types, the informational aspects of the question content and their combinations for evaluation are listed, respectively.

By definition, performance of *Retrieve_M*, *Type_Div*, *Content_Div* and *Type&Content_Div* demonstrate the diverse *infoNeeds* description ability of different

models, namely, the conventional bag-of-word *IR* model, question types, question content and the combination of them based on the results of *IR* model, respectively.

It is observed that *Type_Div* achieves good performance, which consistently ranks in the top 2 best performance among the comparative methods, even only considering the content diversity for evaluation. However, performance of *Content_Div* are comparable with, and even slightly worse than *Retrieve_M*. More detailed analysis should be conducted further to explain this.

Type&Content_Div outperforms in most of the evaluation criteria. This result shows the effectiveness of the proposed recommendation diversification algorithm. This also indicates that correlations exist between the relevant information aspects and question types. Modeling the distribution of these two factors jointly may be helpful to enhance the performance of question recommendations.

Table 2. Question recommendation evaluation performance using *MAP_IA*, *P@5_IA*, *α-nDCG@10*

MAP_IA	Retrieve_M	MDL	MMR	Type_Div	Content_Div	Type&Content_Div
type		0.054	*0.145*	*0.202*	0.094	0.134
content	0.830	0.437	0.778	*0.838*	0.809	*0.899*
type&content	0.297	0.132	0.306	0.309	*0.322*	*0.326*
P@5_IA	Retrieve_M	MDL	MMR	Type_Div	Content_Div	Type&Content_Div
type	0.136	0.091	0.162	*0.229*	0.138	*0.180*
content	0.303	0.189	0.308	*0.372*	0.270	*0.412*
type&content	0.173	0.104	0.189	*0.218*	0.162	*0.256*
α-nDCG@10	Retrieve_M	MDL	MMR	Type_Div	Content_Div	Type&Content_Div
type	0.408	0.288	0.500	*0.591*	0.444	*0.537*
content	0.817	0.494	0.798	0.831	*0.860*	*0.865*
type&content	0.548	0.337	0.597	*0.625*	0.569	*0.663*

MDL-based question recommendation method performs the worst among all the methods. Its poor performance may due to the emphasis of their algorithm. *MDL* algorithm encourages returning questions of which the focus part should have the least similarity with the focus part of the query question. Although this emphasis can return refresh information to users, the usefulness and the diversity of the information are not guaranteed.

The performance of *MMR* and our method are comparable if only considering the question type diversity for evaluation. However, when it comes to the diverse informational aspects in the question content, the performance of *MMR* drops sharply. One possible reason is that this paper use unigram to model the informational aspects. The Euclidean distance based similarity is not sensitive to such small granularity. On the contrary, question types differ from each other not only in the interrogative patterns, but also may have different syntactic structures, thus adding more new information to satisfy the max margin required by *MMR*.

5   Conclusions

This paper proposes a method of question recommendation. The recommended questions are relevant to the query question, and are diverse both in the question type and content. The factors including scores of information retrieval model, importance and novelty of both question types and the informational aspects of question content are combined for question re-ranking. Experimental results demonstrate that the proposed method effectively recommends relevant questions with high information need coverage and diversity in community-base QA.

Overall, performance of diversifying question content is better than that of question types, which shows recommending questions with relevant and diverse *infoNeed* types are essentially more difficult. However, it also indicates large room of future improvement; other effective approaches should be explored later.

References

1. Jiwoon, J., Bruce, C.W., Joon, H.L.: Finding Similar Questions in Large Question and Answer Archives. In: 14th ACM CIKM, Bremen, pp. 84–90 (2005)
2. Robert, S.T.: The Process of Asking questions. American Documentation 13(4), 391–396 (1962)
3. Silvia, Q.: Personalized, Interactive question Answering on the Web. In: Workshop on Knowledge and Reasoning for Answering Questions, Manchester, pp. 33–40 (2008)
4. Burger, J., Cardie, C., Chaudhri, V., Gaizauskas, R., Harabagiu, S., Israel, D., et al.: Issues, Tasks and Program Structures to Roadmap Research in question & Answering (Q&A). Technical Report, NIST. Gaithersburg (2001)
5. Clarke, C.L.A., Craswell, N., Soboroff, I.: Overview of the trec 2009 web track. In: 18th Text Retrieval Conference, NIST (2009)
6. Palakorn, A., Xiaohua, H., Tingting, H.: Answer Diversification for Complex Question Answering on the Web. In: Zaki, M.J., Yu, J.X., Ravindran, B., Pudi, V. (eds.) PAKDD 2010. LNCS, vol. 6118, pp. 375–382. Springer, Heidelberg (2010)
7. Yunbo, C., Huizhong, D., Chin-Yew, L., Yong, Y., Hsiao-Wuen, H.: Recommending Questions Using the MDL-based Tree Cut Model. In: WWW, Beijing, pp. 81–90 (2008)
8. Yaoyun, Z., Xiaolong, W., Xuan, W., Shixi, F.: CogQTaxo: Modeling human cognitive process with a three-dimensional question taxonomy. In: ICMLC, Qingdao, pp. 3305–3310 (2010)
9. Fano, R.: Transmission of Information. MIT Press, Cambridge (1961)
10. Aslam, J.A., Pelekov, E., Rus, D.: The star clustering algorithm for static and dynamic information organization. Journal of Graph Algorithms and Applicatins 8(1), 95–129 (2004)
11. Dekang, L., Patrick, P.: DIRT – Discovery of Inference Rules from Text. In: KDD, San Francisco, pp. 323–328 (2001)
12. Xiaobing, X., Jiwoon, J., Bruce, C.W.: Retrieval models for question and answer archives. In: 31st ACM SIGIR, Singapore, pp. 475–482 (2008)

13. Rakesh, A., Sreenivas, G., Alan, H., Samuel, I.: Diversifying search results. In: 2th ACM WSDM, Barcelona, pp. 5–14 (2009)
14. Clarke, L., Kolla, M., Cormack, G.V., Vechtomova, O., Ashkan, A., Büttcher, S., MacKinnon, I.: Novelty and diversity in information retrieval evaluation. In: 31st ACM SIGIR, Singapore, pp. 659–666 (2008)
15. Carbonell, J., Goldstein, J.: The use of mmr, diversity-based reranking for reordering documents and producing summaries. In: 21st ACM SIGIR, Melbourne, pp. 335–336 (1998)

Classification of Multi-spike Trains and Its Application in Detecting Task Relevant Neural Cliques

Fanxing Hu[1], Bao-Ming Li[2], and Hui Wei[3]

[1,3] Lab of Algorithm for Cognitive Model, School of Computer Science & Technology,
[2] Lab of Higher Brain Functions, Institute of Neurobiology,
Fudan University, Shanghai 201203, China
`hfxfudan@gmail.com,`
`{bmli,weihui}@fudan.edu.cn`

Abstract. Prediction of animal's behavior and detection of task relevant neural cliques using multi-spike trains are of great importance and challenges. We propose a robust and high accurate approach to classify multi-spike trains based on point process model and Bayesian rules. To detect task relevant neural cliques, a graph is constructed with its edge weights indicating the collaboration degree of neurons' trail-to-trail response to tasks. Then minimum graph cut algorithm is introduced to detect neural cliques. Tested by data synchronously recorded in hippocampus during five sets of mouse U maze experiments (about 500 trails), the predicting accuracy is rather high and the statistical significance of the cliques is demonstrated.

Key words: multi-spike trains classification, task relevant cliques, minimum graph cut.

1   Introduction

With the improvements in multi-electrodes synchronous recording techniques, various experiments are designed to study the relationship between neural spike trains and external stimuli [1]. One kind of such experiments which are usually referred as neural decoding is to infer external stimuli that an animal undergoes by analysis of temporal and spatial firing characters of synchronously recorded spikes [1-3]. A frequently used framework is to select most task sensitive neurons whose firing rate is significantly distinguishable among different stimuli or tasks, then HMMs or Bayesian rules are applied to derive post-possibility of the stimuli to an observed spike trains of new coming trails [4-6]. However, in dynamic animal autonomous experiments, neurons recorded esp. in cortex or hippocampus, fires rather variously across trails. This variability may be due to both the randomness of firing mechanism and the deviation in cognitive process on different trails [7]. Therefore, finding a robust method which is tolerant to such variability, works stably in small data set and does not require manual selection of neurons, is appealing and challenging. Based on high order inhomogeneous Poisson model [6, 8], we propose a multi-spike train classification approach, making improvements in spike train transformation algorithm and scoring measure.

Another issue that draws broad attention is to investigate whether or how much a neuron functions in a particular dynamic task. By scrutiny of neurons' relevance to a set of elaborately designed tasks, more precise rolls they play may be inferred [9]. Such a problem is also challenging. Pairwise correlation and JPSTH are two usual ways to analyze neurons' grouping during a task [10, 11]. However, these direct Pearson correlation based techniques are limited to detect multi-scale relationship of neurons. Our basic motivation is that since the cognitive processes in the neuron system vary from trail to trail, if a neuron functions closely in the task, its biological signal that draws neurons firing should has some relationship with this trends. The cognitive process of each trail is unreachable and it is impossible to judge whether the trail-to-trail variability is caused by the cognitive process or by other irrelevant stimuli by just one neuron. However, if a considerable count of neurons response in a correlated way from across trails, such collaboration may be due to the fluctuation of cognitive process. We define neurons' matching value and construct a graph with its edge weights defined as the correlation of neurons' matching values. By iteratively applying minimum edge cut algorithm to cut least relevant neurons, we reserve task relevant ones that forms cliques. To the best of our knowledge, it is the first time that minimum graph cut is introduced in finding neural relationship.

In the following sections, we first present our multi-spike train data classification method and reveal its performance on data recorded in hippocampus during mouse U maze experiments. Then based on correlation matrix of classifying matching values, we introduce minimum graph cuts algorithm to detect task relevant neural cliques and gain each neuron's relevance. The statistical significance of the cliques is demonstrated.

2   Multi-spike Trains Classification

In this section, we reveal the present approach to classify multi-spike trains according to their undergoing tasks. Since trails' durations are not necessarily congruent, spike trains are transformed first. Since the drawing biological signal varies from trail to trail, we model the spiking process with high order inhomogeneous Poisson process. Then we estimate the average point process as their prototypical process. Matching values are derived to measure the possibility that a spike train is generated by certain Poisson process. The final prediction is determined by an integrating strategy of all neurons.

2.1   Spike Train Transformation

In autonomous animal experiments, the animal behaviors are conducted only by animals themselves. In order to estimate neurons' firing rate, spike trains should be transformed into identically long ones. We propose the Recursive Shorten (RS) or Recursive Extend (RE) Algorithm to adjust a spike train's length. Spike train transformation should not destroy the original local or overall statistics including expect and variance, nor should algorithm introduce artificial order.

For programming convenience, RS and RE algorithm uses recursion to clip a spike train into N small pieces, then cut or add a spike train segment of length S/N at a random position. S denotes the length need to be shortened or elongated. For a very small piece of spike train, its firing rate could be regarded as constant. Thus cutting or adding a small segment in the piece would not change its local statistics.
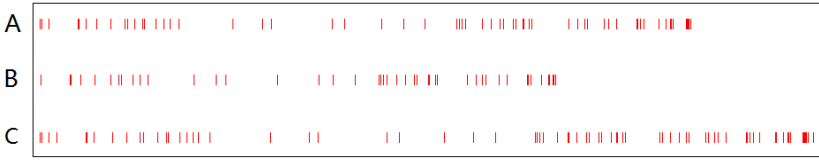
Fig. 1. Tainsformation of spike train. A is the orignal spike train, B is the shortened spike train and C is the extended one.

2.2 Estimating Inhomogeneous Poisson Process

The trail-to-trail cognitive processes of an animal are not identical, consequently leading the biological signal that draws neurons to fire to deviates. We model the spiking process as inhomogeneous process. Since most of the cortex and hippocampus neurons do not fire very fiercely, the effects of burst and refractory period could be ignored. For a single neuron, suppose $\{T_1, T_2, ... T_M, \}$ denotes the spike train set that belongs to the same task. M is the count of trails, and T_i is the ith spike train. Since trail-to-trail inhomogeneous Poisson processes are not identical, $\{\lambda_i(t)\}$ denote the Poisson process parameter for each trail. The average of high order Poisson process parameter $\bar{\lambda}(t)$ is used as the prototypical process. In a sufficient short time bin Δt, non-stationary Poisson process could be regard as stationary process, and $N_i(t) \sim P(\lambda_i(t)\Delta t)$, where $N_i(t)$ is the count of spikes in Δt of i-th spike train, and P denotes Poisson distribution. Because Poisson distribution has summative feature, the sum of spike counts in Δt of all spike trains $\sum N_i(t)$ is subjected to $P(\Delta t \sum \lambda_i(t))$, thus the expect of $\sum N_i(t)$ is $\Delta t \cdot \sum \lambda_i(t)$. Therefore, the estimate of $\bar{\lambda}(t)$ is given by

$$\bar{\lambda}(t) = \frac{\sum_{i=1}^{M} N_i(t)}{M \cdot \Delta t} \tag{1}$$
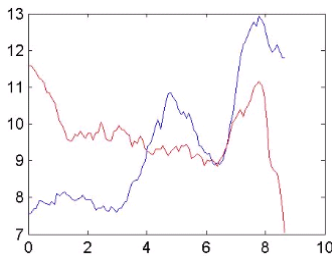
Fig. 2. Inhomogeneous Poisson parameters estimated. This figure reveals estimated $\bar{\lambda}(t)$ of the same neuron of two tasks, using time bin=1 second and 100 estimated points: the blue curve is estimated through 77 spike trains in task 1; the red curve, through 82 trails, in task 2.

Note that equation (1) is the same as the estimate of PSTH, but it provides a different perspective of the process that generates spike trains. It indicates that spike trains are not generated by the same point process across trails, and what we really estimate is the expectation of the high order Poisson point process. Since they have the same form and value, we do not discriminate them in the following sections. At any time t, $\bar{\lambda}(t)$ could be estimated by equation (1). In order to reduce computing complexity, we just estimate some discrete time points and interpret points for the rest. Fig.2 reveals two estimated Poisson curves.

2.3   Bayesian Rules for Matching Value

Suppose $\{\lambda_i^j(t)\}$ denotes the non-stationary Poisson process of the neuron i for task j, and $i \in [1, N], j \in [1, K]$, N is the count of neurons recorded and K is the task count to be recognized. To classify a trail by a single neuron i, it is reasonable to find the largest possibility $p(\lambda_i^j(t)|T_i)$ for any j. T_i denote a new coming trail's spike train of neuron i. A spike train could be decomposed into a series of inter-spike intervals (ISI), $T_i = (ISI_1, ISI_2, ... ISI_S)$ where S is the count of spikes,. According to Bayesian rules,

$$p(\lambda_i^j(t) \mid T_i) = \frac{p(T_i \mid \lambda_i^j(t)) \cdot p(\lambda_i^j(t))}{p(T_i)} \tag{2}$$

For any $k1$, $k2$, $k1 \neq k2$, $p(ISI_{k1}|\lambda_i^j(t))$ and $p(ISI_{k2}|\lambda_i^j(t))$ are mutual independent and $p(T_i)$ is identical for any tasks, thus

$$\max \arg_j p(\lambda_i^j(t) \mid T_i) = p(\lambda_i^j(t)) \cdot \prod_{k=1}^{S} p(ISI_k \mid \lambda_i^j(t)) \tag{3}$$

The $p\left(ISI_k\middle|\lambda_i^j(t)\right)$ obeys the first order waiting time for inhomogeneous Poisson process, thus we have

$$p(ISI_k \mid \lambda_i^j(t)) = \lambda_i^j(t_k) \cdot e^{-\int_{t_{k-1}}^{t_k} \lambda_i^j(\tau)d\tau} \tag{4}$$

where t_k is the occurring time of k-th spike. $p(\lambda_i^j(t))$ is the prior possibility of the task, and for the sake of clarity in equation (4), we simply use p^j denote the possibility of task j. Then, we bring equation (4) into equation (3), and transform equation (3) into its log function,

$$\max \arg_j p(\lambda_i^j(t) \mid T_i) = \log p^j + \sum_{k=1}^{S} \log \lambda_i^j(t) - \sum_{k=1}^{S} \int_{t_{k-1}}^{t_k} \lambda_i^j(\tau)d\tau \tag{5}$$

In equation (5), p^j could be calculated by the total trails counts dividing the occurrence of task j. $\sum_{k=1}^{S} \int_{t_{k-1}}^{t_k} \lambda_i^j(\tau)d\tau$ denote the integration of $\lambda_i^j(\tau)$, and for different spike trains, this value is identical. Thus, equation (5) is rewritten as

$$\max \arg{}_j \ p(\lambda_i^j(t) \mid T_i) = \log p^j + \sum_{k=1}^{S} \log \lambda_i^j(t) - \int_0^t \lambda_i^j(\tau)d\tau \quad (6)$$

Now, the matching value that measures the possibility that a spike train is generated by a known Poisson process has been derived. Note that the most time-consuming terms for computing is the $\int_0^t \lambda_i^j(ô)dô$, and for multiple classification tasks, this term could be computed in advance, thus reducing the computational complexity.

2.4 Integrating Strategy

Through equation (6), we are able to classify trails by single spike train, and obtain their classification $C = (c_1, c_2, ..., c_N)$. The present classifying algorithm include comprehensive analysis of all the neurons' response to reduce errors. Each neuron's classification as well as its confidence of the current classification should be both taken into consideration. For neuron i, $v_i = (v_{i1}, v_{i2}, ..., v_{iK})$ denotes its matching value to task K calculated through equation (6). Suppose $\bar{v}_i$ denotes the average value of v_i, and the confidence of its classification is defined as,

$$conf_i = \frac{1}{1 + \exp(-\max(v_i) + \bar{v}_i)} \quad (7)$$

$max(v_i)$ is the maximum value of v_i. Using sigmoid normalization, each neuron's classifying confidence is between 0 and 0.5. Suppose $conf = (conf_1, conf_2, ..., conf_N)$, then a trail is to be classified into task i, if and only if

$$i = \max \arg{}_j \ \delta_j(C) * conf \quad (8)$$

$ä_i$ function transforms any value that equals i to 1 and others to 0. Equation (8) works similarly to a weighted vote, and the confidence is the weight.

2.5 Multi-Spike Trains Classification Results

To test the performance of the present approach, a U maze experiment is designed in which a mouse is trained to run clockwise and anticlockwise fetching water alternatively. Multi-spike train data and time when the mouse arrive at position 1 and 2 demonstrated in Fig.3(b) are recorded synchronously. Neurons recorded are in mouse's hippocampus by micro-electrode array. Five sets of U maze experiment are carried out in three days, with each set lasting about half an hour. Detailed parameters of the experiment are listed in Tab.1. Due to slight deviation of electrodes and different execution of spike sorting algorithm, counts of sorted neurons among these 5 sets of experiments are not identical. Besides, the mouse did not execute trained task

uninterruptedly and in some trails it stop or even turn back. Therefore, the count of clockwise trails do not equal to the count of anticlockwise trails all the time.

(a)   (b)
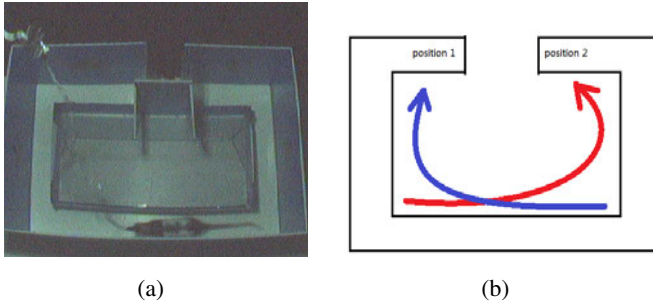
Fig. 3. Demonstration of U maze experiment. Figure (a) is captured from the U maze experiment video, and Figure (b) demonstrates the design of this experiment: mouse is trained to run clockwise and anticlockwise alternatively fetching water. There are two tasks in this experiment: running clockwise (Task 1) and running anticlockwise (Task 2).

10 of left trails and 10 of right trails are selected randomly as train set to estimate neurons' Poisson processes and the rest are used as test set to obtain classification accuracy. This process is repeated 100 times and the average accuracy is regarded as the accuracy of the neuron ensembles. The accuracy is also listed in Tab.1. The classification accuracy is always above 97% in the five set of experiments. Such results provide strong support to the present approach. Moreover, this test also demonstrate the robustness of the present approach obviously, since the train set size is rather small and the neurons recorded are all rough, without refined or selected manually.

Table 1. Detaild parameters and classification results of U maze experiment

No.	Neuron	Clockwise	Anticlockwise	Duration	Accuracy
1	25	77	82	1680 sec	97.94 %
2	37	43	47	1798 sec	98.53 %
3	33	60	60	1789 sec	99.82 %
4	25	24	23	1080 sec	98.55 %
5	30	24	23	1364 sec	99.10 %

To investigate the classification results closely, we reveals each neuton's accuracy under 10 and 20 count of train trails in Fig.4. It is noticable that the increase in the size of train trails contribute little in improvement of classification accuracy. Such a results illustrate that through 10 trails the estimated Poisson process is steady enough to get a good classification result. The prominent improvement of neural classification lies in the integrating stratage of neurons, and such a phoneminen confirms the importance in population encoding of neural system.
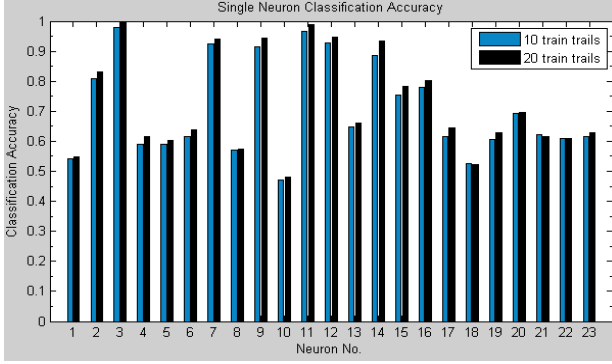
Fig. 4. This figure reveals single neuron classification accuracy by 10 train trails and 20 train trails in the first set of data. The blue bar represent the accuracy by 10 train trails, and the black bar is for 20 train trail. This figure illustrates that the increase of train set improves little in classification accuracy.

3   Detection of Task Relevant Neural Cliques

The trail-to-trail variability of the cognitive process and the random feature of spiking process have been considered to be the main cause of variability across trails. The variability of spiking process under Poisson model could be measured by Fano factor. For Poisson model the Fano factor should be 1. However, in most of experiments, especially in autonomous animal experiments, the observed Fano factor is far larger than 1, which illustrates that most part of the trail-to-trail variability is caused by the fluctuation of the underlying biological signal. And if a neural ensemble functions during a particular task, some of the neurons' biological signal varies relevantly with the cognitive process from trail to trail. By detecting the correlation of neurons' response, we could find the task relevant cliques.

First, we find a prototypical process for each neuron's response to a task and as is discussed above, a proper prototypical is neuron's average inhomogeneous Poisson process which could be estimated by equation (1).   Then a measurement should be defined to compare the distance between a trail and a point process. The measurement is also discussed above, and the log-possibility in equation (6) performs well. Suppose $M_i = (m_{i1}, m_{i2}, ..., m_{iT})$ denote the matching value of all trails of a task of neuron i. Then we calculate the pairwise correlation matrix of $\{M_i\}$, and $\{r_{ij}\}$ is the Pearson correlation of M_i and M_j. The Pearson correlation is given by

$$ r_{XY} = \frac{E((X - \bar{X})(Y - \bar{Y}))}{\sqrt{D(X)D(Y)}} \tag{9} $$

Then we construct a correlation graph whose vertexes represent neurons and edges are their correlation coefficients r_{ij}. In order to detect the cliques, minimum graph cut is applied iteratively until it reaches a user defined threshold. The pseudo code is listed below in matlab style.

```
function V = cliques(G, threshold)
  V = {};
  P = {G};
  while(P is not empty)
    take an element in P as G;
    Remove G from P;
    if minAveDegree(G)>threshold
        Put G into V;
    else
        G.degree = Degree(G);
        [G1 G2]= mincut(G);
        G1.degree= G.degree + Degree(G1)*G1.size/G.size;
        G2.degree= G.degree + Degree(G2)*G2.size/G.size;
        put G1, G2 into P;
    end
  end
end
```

The function minAveDegree() returns the minimum average degree of vertex in a graph. This criterion for cutting a graph guarantees the detected cliques guarantees even the least correlated vertexes meet the given threshold. The function mincut() cuts a graph into two sub-graphs using minimum cut edges, we implement this algorithm according to [12]. The function Degree() returns the mean degree of a graph. By iteratively cutting least relevant neurons and the remaining cliques are highly coherent and task relevant.

(a)   (b)
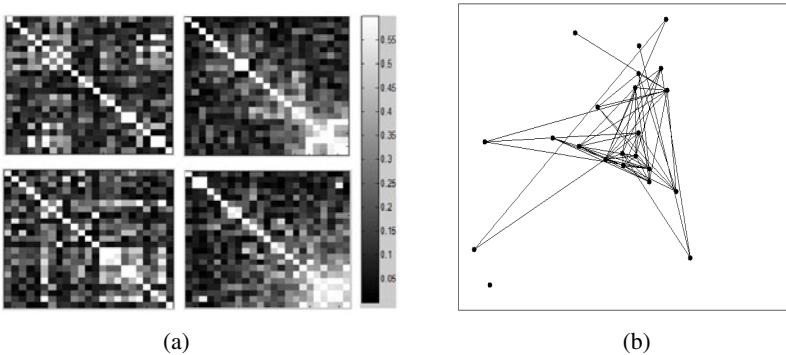
Fig. 5. This figure reveals the gathering effect using degrees to sort neurons. In figure (a), the top left and bottom left figures are the original correlation matrixes of matching value in task 1 (clockwise) and task 2 (anticlockwise) respectively, and the top right and bottom figures are the sorted matrixes. Figure (b) reveals the arranged connectivity under 0.01 level significance of task 1. The deviations of neurons to the center accord with their reciprocals of relevance.

The algorithm also derives degrees indicating the relevance of a neuron to a task. The degrees of the observed neurons map pairwise relationship into topological relationship by their collaboration level with others. Fig.5(a) reveals the comparison of original correlation matrix and the sorted correlation matrix using neurons' degree. Obviously, the correlated neurons gather in the right bottom corner of the graph while others gather in the opposite direction. Fig.5(b) is the connectivity graph arranged by neurons' relevance to a task.

In order to illustrate the correlation of matching value between neurons has the statistical significance. We disorder each neuron's matching value respectively and then apply the present algorithm again. Fig.6 reveals the results and degrees of neurons calculated in random task are conspicuously lower than the real ones.

Table 2. Cliques Derived by Minimun Graph Cut

No.		Cliques
1	Task 1	{7, 8, 15, 19, 22}, {2, 3, 5}
	Task 2	{2, 6, 10, 14, 19}, {8, 13}, {11, 20}
2	Task 1	{5, 20, 8, 19, 9, 18, 11}, {14, 13}, {24, 23}
	Task 2	{4, 2, 20, 11}, {17, 19}, {10, 3, 21}, {15, 14}
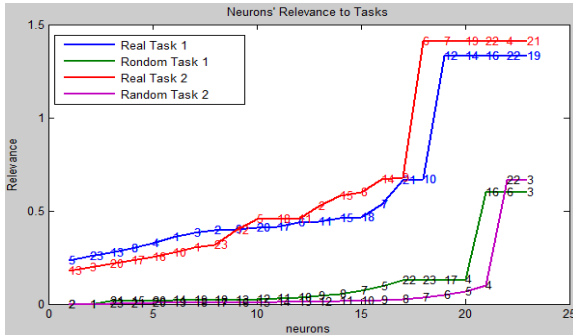
Fig. 6. This figure reveals the degrees of neurons relevance of in real task and random tasks. The blue and red curves are the neuron's degree in task 1 and task 2 respectively and the brown and purple curves are derived from disordered matching value.

In addition, we adjust the algorithm to obtain cliques with grounded statistical significance using the 0.05 significance level null hypnosis instead of $\{r_{ij}\}$. Cliques detected in 2 sets of U maze experiment with the two tasks are listed in Tab.2.

4   Discussion

We have presented our method to classify trails by multi-spike train data. Based on high order inhomogeneous Poisson model and Bayesian rules, the classification method performs well in several aspects: (1) the accuracy of classification is high, since in about 500 trails prediction test, the accuracy is nearly 99%; (2) the train set is rather small, only random 10 train trails for each task in the U maze experiment is needed to obtain remarkable results; (3) No manual work have done to select rough recorded data. Besides, we introduce a novel approach to analyze the task relevant neural dependencies based on neurons' matching value which indicating their distance to the prototypical Poisson processes. Minimum graph cut is proposed to derive task relevant neural cliques which are significant in statistics. Further work should be done to find evidence that such cliques may have close biological connectivity or act in concert to fulfill certain function.

References

1. Brown, E.N., Kass, R.E., Mitra, P.P.: Multiple neural spike train data analysis: state-of-the-art and future challenges. Nature Neuroscience 7, 456–461 (2004)
2. Bialek, W., Rieke, F., De Ruyter Van Steveninck, R.R., Warland, D.: Reading a neural code. Science 252, 1854 (1991)
3. Wilson, M.A., McNaughton, B.L.: Dynamics of the hippocampal ensemble code for space. Science 261, 1055 (1993)
4. Brown, E.N., Frank, L.M., Tang, D., Quirk, M.C., Wilson, M.A.: A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells. The Journal of Neuroscience 18, 7411 (1998)
5. Eden, U.T., Frank, L.M., Barbieri, R., Solo, V., Brown, E.N.: Dynamic analysis of neural encoding by point process adaptive filtering. Neural Computation 16, 971–998 (2004)
6. Heeger, D.: Poisson model of spike generation. Handout, University of Standford 5 (2000)
7. Cunningham, J.P., Yu, B.M., Shenoy, K.V., Sahani, M.: Inferring neural firing rates from spike trains using Gaussian processes. Advances in neural information processing systems 20, C329–C336 (2008)
8. Katahira, K., Nishikawa, J., Okanoya, K., Okada, M.: Extracting state transition dynamics from multiple spike trains with correlated Poisson HMM. Advances in Neural Information Processing Systems 21, 817–824 (2009)
9. Lin, L., Osan, R., Tsien, J.Z.: Organizing principles of real-time memory encoding: neural clique assemblies and universal neural codes. TRENDS in Neurosciences 29, 48–57 (2006)
10. Ito, H., Tsuji, S.: Model dependence in quantification of spike interdependence by joint peri-stimulus time histogram. Neural Computation 12, 195–217 (2000)
11. Bertoni, A., Campadelli, P., Grossi, G.: A neural algorithm for the maximum clique problem: analysis, experiments, and circuit implementation. Algorithmica 33, 71–88 (2002)
12. Stoer, M., Wagner, F.: A simple min-cut algorithm. Journal of the ACM (JACM) 44, 585–591 (1997)

Dreaming Your Fear Away: A Computational Model for Fear Extinction Learning during Dreaming

Jan Treur

VU University Amsterdam, Agent Systems Research Group
De Boelelaan 1081, 1081 HV, Amsterdam, The Netherlands
`treur@cs.vu.nl`
`http://www.cs.vu.nl/~treur`

Abstract. In this paper a computational model is presented that models how dreaming is used to learn fear extinction. The approach addresses dreaming as internal simulation incorporating memory elements in the form of sensory representations and their associated fear. During dream episodes regulation of fear takes place, which is strengthened by Hebbian learning. The model was evaluated by a number of simulation experiments for different scenarios.

Keywords: dreaming, fear extinction learning, computational model.

1   Introduction

In the recent cognitive and neurological literature the mechanisms and functions of dreaming have received much attention; e.g., [19-23], [27-32]. In such literature, usually dreaming is considered a form of internal simulation of real-life-like processes serving as training in order to learn or adapt certain capabilities. Dreaming makes use of memory elements for sensory representations (mental images) and their associated emotions to generate 'virtual simulations'; e.g., [20], pp. 499-500. Taking into account fear emotions that often play an important role in dreams, strengthening of regulation of such emotions is considered an important purpose of dreaming; see, for example, [20, 30]. To this end in dreams adequate exercising material is needed: sensory representations of emotion-loaden situations are activated, built on memory elements suitable for high levels of arousal:

> 'They are recombined or remapped in order to introduce elements that are incompatible with existing fear memories, thus facilitating (among other functions) the acquisition or maintenance of extinction memories. The latter inhibit fear memories (..), and consequently alleviate affect load.' ([20], pp. 500-501)

A comparison can be made to a virtual reality form of exposure therapy ([20], pp. 500-501). Strong fear associations of the sensory memory elements used to make up a dream creates situations in which a person has to cope with high levels of fear. Adopting basic elements from [26] the computational model presented here generates the type of internal simulation that is assumed to take place in dreaming. For the different dream episodes, the internal simulation incorporates interrelated processes of activation of sensory representation states (from memory) providing mental images,

and activation of associated feelings. Moreover, it incorporates emotion regulation to suppress the feeling levels and the sensory representation states. The regulation mechanism strengthens the relevant connections by Hebbian learning; e.g., [2, 10, 16].

The structure of the paper is as follows. In Section 2 the computational model is described in more detail. Section 3 presents simulation results providing some dream scenarios. Finally, Section 4 is a discussion, in which also the relation of the model with neurological theories and findings is addressed.

2   A Computational Model for Fear Extinction Learning

The computational model presented here is based on mechanisms suggested in neurological literature; see Fig. 1 for an overview of the states and connections. Some of the (non-adaptive) basic elements were adopted from [26]. In Fig. 1 the basic model for a given stimulus s_k with sensory representation state srs_{s_k} and dream episode state es_{s_k} is shown ($k = 1, ..., n$). An explanation of the states used is shown in Table 1; an overview of the connections is shown in Table 2. Note that in Fig. 1 a sensory representation state and episode state for only one stimulus s_k is depicted. In the specification of the model below an arbitrary number n of such states are taken into account. In Section 4, a simulation scenario with four stimuli s_k is presented.
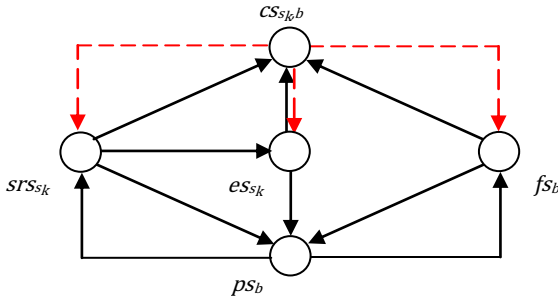
Fig. 1. Overview of the states and connections in the model

The inhibiting links for fear regulation are indicated by dotted arrows (in red). The two links between srs_{s_k} and ps_b indicate the bidirectional association between stimulus s_k and emotional response b. The links between ps_b and fs_b indicate a recursive as-if body loop (see below).

Table 1. Overview of the state variables used

state	explanation
ps_b	Preparation state for bodily response b
fs_b	Feeling state for b
srs_{s_k}	Sensory representation state for stimulus s_k
$cs_{s_k,b}$	Control state for regulation of sensory representation of s_k and feeling b
es_{s_k}	Dream episode state for s_k
mt_{s_k}	Memory trigger for s_k

Table 2. Overview of connections and weights

from states	to state	weights	LP
$srs_{s_1}, ..., srs_{s_n}, fs_b, es_{s_1}, ..., es_{s_n}$	ps_b	$\omega_{1,1}, ... \omega_{1,n}, \omega_2, \omega_{12,1}, ... \omega_{12,n}$	LP1
$ps_b, cs_{s_1,b}, ..., cs_{s_n,b}$	fs_b	$\omega_3, \omega_{4,1}, ...\omega_{4,n}$	LP2
$ps_b, cs_{s_k,b}, mt_{s_k}$	srs_{s_k}	$\omega_{5,k}, \omega_{6,k}, \omega_{0,k}$	LP3
$srs_{s_k}, fs_b, es_{s_k}$	$cs_{s_k,b}$	$\omega_{7,k}, \omega_{8,k}, \omega_{13,k}$	LP4
$srs_{s_k}, es_{s_1}, ..., es_{s_n}, cs_{s_k,b}$	es_{s_k}	$\omega_{9,k}, \omega_{10,1,k}, ..., \omega_{10,n,k}, \omega_{11,k}$	LP5

The model incorporates four connected cycles (see Fig. 1):

➢ A positive preparation-feeling cycle $ps_b – fs_b$ (right lower part in Fig. 1)

➢ A positive preparation-sensory representation cycle $ps_b – srs_{s_k}$ (left lower part)

➢ A negative emotion regulation cycle $cs_{s_k,b} – fs_b, srs_{s_k}, es_{s_k}$ (upper part)

➢ A positive fear extinction learning cycle $cs_{s_k,b} – \omega_{7,k}, \omega_{8,k}$ (upper part)

Each of these cycles will be briefly discussed.

The Preparation-Feeling Cycle $ps_b – fs_b$

As indicated in Section 1 above, dreams can be considered as flows of activated imaginations based on (re)combined sensory memory elements with emotional associations. Such flows can be related to the notion of *internal simulation* put forward, among others, by [4, 5, 12, 17, 18]. The idea of internal simulation is that sensory representation states are activated (e.g., mental images), which in response trigger associated preparation states for actions or bodily changes, which, by prediction links, in turn activate other sensory representation states.

    sensory representation states $\rightarrow$ preparation states $\rightarrow$ sensory representation states

Internal simulation has been used, for example, to describe prediction of effects of own actions (e.g., [3]), processes in another person's mind (e.g., [12]) or processes in a person's own body (e.g., [4]). The idea of internal simulation has been exploited in particular by applying it to bodily changes expressing emotions, using the notion of *as-if body loop* (cf. [4], pp. 155-158; [5], pp. 79-80; [7]):

    sensory representation $\rightarrow$ preparation for bodily changes = emotional response $\rightarrow$
    emotion felt = based on sensory representation of (simulated) bodily changes

Damasio [4] distinguishes an emotion (or emotional response) from a feeling (or felt emotion). The emotion and feeling in principle mutually affect each other in a bidirectional manner: an as-if body loop usually occurs in a recursive, cyclic form by assuming that the emotion felt in turn also affects the prepared bodily changes, as he points out, for example, in ([6], pp. 91-92; [7], pp. 119-122):

    emotion felt = based on sensory representation of (simulated) bodily changes $\rightarrow$
    preparation for bodily changes = emotional response

The Preparation-Sensory Representation Cycle ps_b – srs_{sk}

Sensory representations as stored in memory usually have emotional responses associated to them. This means that as soon as a sensory representation is activated also its associated emotional response preparations are activated, and, conversely, when an emotional response preparation is active, also the sensory representations associated to this type of response become active. This results in a cycle between sensory representations srs_{sk} and emotional response preparations ps_b shown in the left lower part of Fig. 1. Together with the preparation – feeling cycle discussed above, this provides a state of fear as a complex and cyclic activation state of fear response preparations, fear feelings and fearful sensory representations.

The Emotion Regulation Cycle cs_{skb} – fs_b, srs_{sk}, es_{sk}

Fear extinction indicates the process of suppressing fear states. This can be considered a specific type of *emotion regulation* to control emotions that are felt as too strong; cf. [11, 13, 14]. Emotion regulation mechanisms cover *antecedent-focused regulation* (e.g., selection and modification of the situation, attentional deployment, and reappraisal) and *response-focused regulation* (suppression of a response). Regulation of high levels of fear can take place by antecedent-focused emotion regulation, for example, by *attentional deployment* in the form of focusing attention in such a way that situations or aspects of situations in which too strong fear-related stimuli occur are kept out of the attention focus, or by a form of *re-appraisal* decreasing the negative feeling level based on changing the cognitive interpretation of fear-related stimuli into a less negative one. In the upper part of Fig. 1 such an emotion regulation mechanism is depicted. The upward arrows to the control state cs_{skb} take care for monitoring the sensory representations srs_{sk}, feeling state fs_b and episode state es_{sk} for the fear state, and when the fear level is too high, this leads to activation of the relevant control states cs_{skb}. These control states in turn lead to inhibition of the fear-related states (the downward, dotted arrows in the upper part of Fig. 1).

The Fear Extinction Learning Cycle cs_{skb} – $\omega_{7,k}$, $\omega_{8,k}$

The basis of fear extinction learning is that the emotion regulation mechanisms discussed above are adaptive: they are strenghtened over time when they are intensively used. Note that fear extinction learning is *not* a form of unlearning or extinction of acquired fear associations, but it is additional *learning of fear inhibition* in order to counterbalance the fear associations which themselves remain intact (e.g., [20], p. 507). This learning process is modelled by applying a Hebbian learning principle (e.g., [2, 10, 16]) to the upward connections $\omega_{7,k}$ and $\omega_{8,k}$ from sensory representation state srs_{sk} and feeling state fs_b to the control state cs_{skb} in the upper part of Fig. 1. Note that the dream episode state and its upward link to the control state serve as an amplifier in this Hebbian learning process. The positive cyclic character of this learning process is as follows: the stronger the upward connections become, the higher the activation level of the control state, and this again strengthens the learning process for the connections.

The computational model has been formalised as a set of differential equations. Parameter γ is used as a speed factor, indicating the speed by which an activation level is updated upon received input from other states. During processing, each state has an activation level represented by a real number between *0* and *1*. Below, the

(temporally) Local Properties (LP) for the dynamics based on the connections between the states in Fig. 1 are described by differential equations. In these specifications a threshold function *th* is used as a combination function for *k* incoming connections as follows: the combined input level is $th(\mu_1 V_1 + ... + \mu_k V_k)$ with μ_i the connection strength for incoming connection *i* and V_i the activation level of the corresponding connected state. For this threshold function *th* different choices can be made. In the simulation experiments (in LP1 to LP4) the following continuous logistic form was used:

$$th(X) = (\frac{1}{1 + e^{-\sigma(X - \tau)}} - \frac{1}{1 + e^{\sigma\tau}})(1 + e^{-\sigma\tau}) \tag{1}$$

Here σ is a steepness and τ a threshold parameter. Note that for higher values of $\sigma\tau$ (e.g., σ higher than $20/\tau$) this threshold function can be approximated by the simpler expression; this has been used in LP5:

$$th(X) = \frac{1}{1 + e^{-\sigma(X - \tau)}} \tag{2}$$

The first property LP1 describes how preparation for response *b* is affected by the sensory representation and episode states of stimuli s_k (triggering the response), and by the feeling state for *b*:

LP1 Preparation State for Response *b*

$$\frac{dps_b(t)}{dt} = \gamma \, [\, th(\Sigma_k \omega_{1,k} srs_{s_k}(t) + \omega_2 \, fs_b(t) + \Sigma_k \omega_{12,k} es_{s_k}(t)) - ps_b(t)] \tag{3}$$

The feeling state for *b* is not only affected by a corresponding preparation state for *b*, but also by the inhibiting control states for s_k and *b*. This is expressed in dynamic property LP2. Note that for this suppressing effect the connection weight $\omega_{4,k}$ from the control state for s_k and *b* to feeling state for *b* is taken negative, for example $\omega_{4k} = -1$.

LP2 Feeling State for *b*

$$\frac{dfs_b(t)}{dt} = \gamma \, [\, th(\omega_3 \, ps_b(t) + \Sigma_k \, \omega_{4,k} \, cs_{s_k, b}(t)) - fs_b(t) \,] \tag{4}$$

The sensory representation state for s_k is affected by the preparation state for *b* (fear association) and by the suppressing control state for s_k and *b*. For this suppressing effect the connection weight ω_{6k} from the control state for s_k and *b* is taken negative. This is expressed in dynamic property LP3. Moreover, property LP3 is used to describe how the sensory representation of any traumatic s_k is triggered from memory, as a starting point for a dream: in a scenario the memory trigger values are taken *1*. For non-traumatic s_k such triggering does not take place: the values are taken *0*.

LP3 Sensory Representation State for s_k

$$\frac{d \, srs_{s_k}(t)}{dt} = \gamma \, [\, th(\omega_{5,k} \, ps_b(t) + \omega_{6,k} \, cs_{s_k b}(t) + \omega_{0,k} \, mt_{s_k}(t)) - srs_{s_k}(t) \,] \tag{5}$$

Activation of a control state for a specific sensory representation for s_k and *b* is based on the level for feeling *b* and the activation level of the sensory representation and episode states of s_k:

LP4 Control State for s_k and b

$$\frac{d\,cs_{sk,b}(t)}{dt} = \gamma\,[\,th(\omega_{7,k}\,srs_{sk}(t) + \omega_{8,k}\,fs_b(t) + \omega_{13,k}\,es_{sk}(t)) - cs_{sk,b}(t)] \tag{6}$$

Due to the inherent parallellism in neural processes, at each point in time multiple sensory representation states can be active simultaneously. For cases of awake functioning the *Global Workspace Theory* (e.g., [1]) was developed to describe how a single flow of conscious experience can come out of such a large multiplicity of (unconscious) parallel processes. The basic idea is that based on the various unconscious processes a *winner-takes-it-all competition* takes place to determine which one will get dominance and be included in the single flow of consciousness. This idea was applied here in the dreaming context to determine which sensory representation element will be included as an episode state es_{sk} in a dream. This competition process is decribed in LP5, using mutual inhibiting connections from episode states es_{si} with $i \neq k$ to es_{sk}. For the suppressing effects the connection weights from the es_{si} with $i \neq k$ to es_{sk} are taken negative, for example $\omega_{10,i,k} = -0.6$ for $i \neq k$. Note that for the sake of notational simplicity $\omega_{10,k,k} = 0$ is taken. For traumatic stimuli s_k an additional and strong way of inhibition of the corresponding episode state takes place, blocking the generation of an episode state for this stimulus. It is based on the control state for s_k and b and is assumed to have a strong negative connection strength $\omega_{11,k}$. For non-traumatic stimuli this connection is given strength *0*.

LP5 Episode State for s_k

$$\frac{d\,es_{sk}(t)}{dt} = \gamma\,[\,th(\omega_{9,k}\,srs_{sk}(t) + \omega_{11,k}\,cs_{sk,b}(t) + \Sigma_i\,\omega_{10,i,k}\,es_{si}(t)) - es_{sk}(t)] \tag{7}$$

Hebbian Learning to Strengthen Fear Extinction
From a Hebbian perspective [16], strengthening of a connection over time may take place when both nodes are often active simultaneously ('neurons that fire together wire together'). The principle goes back to Hebb [16], but has recently gained enhanced interest by more extensive empirical support (e.g., [2]), and more advanced mathematical formulations (e.g., [10]). In the adaptive computational model two upward connections that play a role in monitoring for the emotion regulation cycle are adapted based on a Hebbian learning principle. More specifically, for such a connection from node i to node j its strength ω_{ij} is adapted using the following *Hebbian learning rule*, taking into account a maximal connection strength *1*, a *learning rate η*, and an *extinction rate ζ* (usually taken small):

$$\frac{d\,\omega_{ij}(t)}{dt} = \eta\,a_i(t)a_j(t)(1 - \omega_{ij}(t)) - \zeta\omega_{ij}(t) = \eta\,a_i(t)a_j(t) - (\eta\,a_i(t)a_j(t) + \zeta)\,\omega_{ij}(t) \tag{8}$$

Here $a_i(t)$ and $a_j(t)$ are the activation levels of node i and j at time t and $\omega_{ij}(t)$ is the strength of the connection from node i to node j at time t. A similar Hebbian learning rule can be found in [10], p. 406. By the factor $1 - \omega_{ij}(t)$ the learning rule keeps the level of $\omega_{ij}(t)$ bounded by *1* (which could be replaced by any other positive number); Hebbian learning without such a bound usually provides instability. When the extinction rate is relatively low, the upward changes during learning are proportional to both $a_i(t)$ and $a_j(t)$ and maximal learning takes place when both are *1*. Whenever

one of $a_i(t)$ and $a_j(t)$ is 0 (or close to 0) extinction takes over, and ω_{ij} slowly decreases (unlearning). This learning principle has been applied (simultaneously) to the two upward connections from sensory representation and feeling states to the control state in Fig. 1, according to the following instantiations of the general learning rule above:

$$\frac{d\omega_{7,k(t)}}{dt} = \eta\, srs_{sk}(t)\, cs_{skb}(t)(1 - \omega_{7,k}(t)) - \zeta\omega_{7,k}(t)$$
$$= \eta\, srs_{sk}(t)\, cs_{skb}(t) - (\eta srs_{sk}(t)\, cs_{skb}(t) + \zeta)\, \omega_{7,k}(t) \qquad (9)$$

$$\frac{d\omega_{8,k(t)}}{dt} = \eta\, fs_b(t)\, cs_{skb}(t)(1 - \omega_{8,k}(t)) - \zeta\omega_{8,k}(t)$$
$$= \eta\, fs_b(t)\, cs_{skb}(t) - (\eta\, fs_b(t)\, cs_{skb}(t) + \zeta)\, \omega_{8,k}(t) \qquad (10)$$

In principle, the learning rate η and extinction rate ζ, can be taken differently for the different connections. In the example simulations discussed in Section 4 (shown in Fig. 2) the following values have been used: $\eta = 0.7$ for all $\omega_{7,k}$ and $\eta = 0.4$ for all $\omega_{8,k}$, and $\zeta = 0.001$ for all $\omega_{7,k}$ and $\omega_{8,k}$.

4   Simulations of Fear Extinction Learning in Dream Scenarios

In dream scenarios in which the cycles as discussed play their roles as follows.

Triggering s_1
- A stimulus s_1 is given for which previously a high extent of fear has been experienced, and for which from time to time (in particular during sleep) a sensory representation state is triggered by memory (for the model this is considered an external trigger); note that such a memory trigger was not used for the other stimuli: their activation automatically happens due to the high fear levels induced by triggering s_1, and maintained by the subsequent dream episisies.
- The activation of the sensory representation of s_1 leads to activation of an enhanced preparation level for a bodily fear response b

The positive preparation-feeling cycle ps_b – fs_b
- By an as-if body loop an enhanced preparation level for b leads to an enhanced fear feeling level for b and vice versa

Blocking s_1
- By a strong form of emotion regulation in particular the sensory representation and episode state of s_1 are strongly suppressed: the activation level of the sensory representation of s_1 becomes low, and no dream episode state for s_1 occurs, as this is blocked

The positive preparation-sensory representation cycle ps_b – srs_{sk}
- Other fear-associated stimuli s_k for $k \geq 2$ are available for which the person has less strong previous experiences; the sensory representation states for these s_k are activated by links from the high preparation state for b, depending on the strength of these links
- When the sensory representation state of a stimulus s_k is activated, this leads to an enhanced activation level of the preparation state for the emotional fear response

The positive preparation-feeling cycle ps_b – fs_b
- Due to the higher activation level of preparation for fear based on b, via the as-if body loop also the feeling level for b becomes higher: the person experiences more fear

Competition to achieve a dream episode es_{sk}
- The active sensory representation for some s_k leads to a corresponding dream episode state, according to a competion process by mutual inhibition to get dominance in the episode

The negative emotion regulation cycle $cs_{s_k,b} - fs_b, srs_{s_k}, es_{s_k}$
- By the control states for emotion regulation for an active sensory representation for s_k both the fear feeling level and the sensory activation level of s_k are suppressed (resp., re-appraisal, attentional deployment)
.
The fear extinction learning cycle $cs_{s_k,b} - \omega_{7,k}, \omega_{8,k}$
- Due to nonzero activation levels of the control states and the fear feeling state for b, and the sensory representation and episode states for s_k Hebbian learning takes place strengthening the connections from feeling state and sensory representation to control state
- Increased connection strengths lead to higher activation levels for the control states

A variety of simulation experiments have been performed according to such scenarios, using numerical software. In the experiments discussed below (see Fig. 2) the settings were as shown in Table 3.

Table 3. Settings used for connection strength, threshold and steepness parameters

from state	connection		to state	threshold	steepness
srs_{s_k}	$\omega_{1,k}$	1			
fs_b	ω_2	1	ps_b	0.5	4
es_{s_k}	$\omega_{12,k}$	1			
ps_b	ω_3	1	fs_b	0.5	4
$cs_{s_k,b}$	$\omega_{4,k}$	-0.2			
ps_b	$\omega_{5,1}$	0.5	srs_{s_1}	0.25	8
$cs_{s_1,b}$	$\omega_{6,1}$	-2			
ps_b	$\omega_{5,2}$	0.5	srs_{s_2}	0.25	8
$cs_{s_2,b}$	$\omega_{6,2}$	-0.5			
ps_b	$\omega_{5,3}$	0.45	srs_{s_3}	0.25	8
$cs_{s_3,b}$	$\omega_{6,3}$	-0.5			
ps_b	$\omega_{5,4}$	0.4	srs_{s_4}	0.25	8
$cs_{s_4,b}$	$\omega_{6,4}$	-0.5			
srs_{s_1}	$\omega_{7,1}$	1			
fs_b	$\omega_{8,1}$	1	$cs_{s_k,b}$	1	8
es_{s_k}	$\omega_{13,k}$	0.3			
srs_{s_k}	$\omega_{9,k}$	1			
es_{s_j} $(j \neq k)$	$\omega_{10,j,k}$	-0.6	es_{s_k}	0.25	60
$cs_{s_k,b}$ $(k \geq 2)$	$\omega_{11,k}$	-0.2			
$cs_{s_1,b}$	$\omega_{11,1}$	-20			

As shown in the left hand side of the table, all noninhibiting connections to preparation, feeling, control, and episode states have strength *1*, and all inhibiting connections from control states to feeling, sensory representation states and episode states, and mutually between episode states have strengths *-0.2*, *-0.5*, *-0.2*, and *-0.6*, respectively, with an exception for the sensory representation and episode states for s_1, which are inhibited by strength *-2* and *-20* (they are blocked due to a previous traumatic event involving s_1). Small differences in emotional associations for the different s_k are expressed by different strengths from preparation of emotional response to sensory representation states, varying from *0.5* to *0.4*. In the scenarios
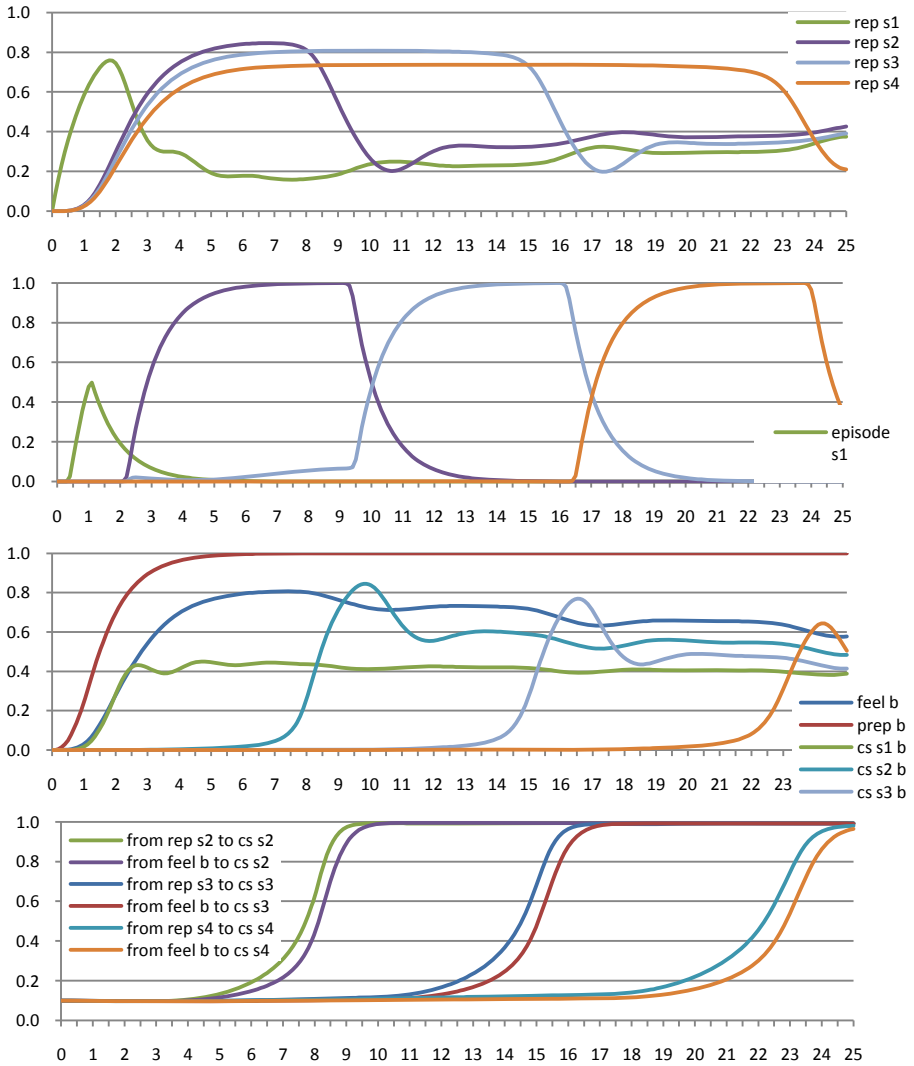
Fig. 2. Dream with three episodes showing extinction learning and reduction of feeling level

considered, the memory trigger for the sensory representation of s_1 has level 1 and connection strength 0.5. The threshold and steepness values used are shown in the right hand side of Table 3. Relatively low steepness values were used, except for the episode states. The threshold values for preparation and feeling states were taken 0.5; in order to model differences in emotional associations between the s_k, different threshold values were taken for their sensory representation and control states. The initial values of all states were taken 0, and for the adaptive connection strengths 0.1

initially (which also could be taken 0). The speed factor γ was 1, and the step size Δt was taken 0.1. For learning and extinction rates the following values have been used: $\eta = 0.7$ for all $\omega_{7,k}$ and $\eta = 0.4$ for all $\omega_{8,k}$, and $\zeta = 0.001$ for all $\omega_{7,k}$ and $\omega_{8,k}$. The example scenario discussed addresses a case where three dream episodes occur, related to the sensory representations of s_2, s_3, s_4, subsequently. In Fig. 2 time is on the horizontal axis and the activation levels of the indicated states and connections are on the vertical axis. In the first graph it is shown that right from the start the sensory representation for s_1 becomes active (triggered from memory). Immediately the emotional response preparation for b starts to develop, and the related feeling, as shown in the third graph. Also in the third graph it is shown how as a result the control state for s_1 becomes active. Due to the strong suppression, no (full) dream episode develops for s_1, as shown in the second graph. Due to the relatively high emotional response and feeling level, the sensory representations for s_2, s_3, s_4 become active, following that order and strength (first graph).

In a cyclic process, this further increases the emotional response preparation and feeling levels (third graph). As the sensory representation of s_2 is the strongest, it wins the competition for the dream episode from time point 3 to 9 (second graph).

Given this first episode and the high feeling and sensory representation levels, extinction learning takes place of the connections to the control state for s_2 (see fourth graph), reaching strengths one around 1 at time point 9, and hand in hand with this process the level of the control state for s_2 jumps up from time point 7 on (see third graph). As a result of this, control is exerted, suppressing after time point 9 the feeling level (third graph), the sensory representation of s_2 (first graph), and the related episode (second graph). As the feeling level was only partly reduced, and the sensory representation for s_2 does not compete anymore, from time point 11 on a second episode occurs, based on the sensory representation of s_3 (second graph). Again the whole adaptation process occurs, this time related to s_3. From time point 16 on, this brings the feeling level more down (third graph), and suppresses the sensory representation of s_3 (first graph), and the related episode (second graph). After this, the whole process repeats iteself for a third dream episode, based on the sensory representation of s_4. This leads to another reduction of the feeling level around time 25. Overall, all connections for fear extinction in relation to the most strongly fear-related sensory representations have been learned and have values around 1, and the feeling level was reduced to below 0.6.

4   Discussion

The assumption that dreaming, especially when negative emotions are involved, can be considered as a purposeful form of internal simulation is widely supported, in particular, for the purpose of strengthening fear emotion regulation capabilities; cf. [9, 15, 20, 21, 29, 30, 32]. In this paper a computational model was presented that models the generation of dream episodes from an internal simulation perspective, and uses these episodes for fear extinction learning. Building blocks to create such internal simulations are memory elements in the form of sensory representations and their associated emotions. The model exploits a mutual (winner-takes-it-all) competition

process to determine sensory representation states that dominate in different dream episodes, comparable to one of the central ideas underlying the Global Workspace Theory of consciousness (cf. [1]). Adaptive emotion regulation mechanisms (cf. [11, 13, 14]) were incorporated to regulate the activation levels of the feeling (by re-appraisal) and the sensory representation states (by attentional deployment). Adaptation in the model is based on Hebbian learning. The computational model was evaluated by a number of simulation experiments for scenarios with different numbers of dream episodes.

In [20] dreaming is related to a network of four main brain components (called the AMPHAC network) and their connections: Amygdala, Medial PreFrontal Cortex (MPFC), Hippocampus, Anterior Cingulate Cortex (ACC). Note that the biological counterparts of the preparation and sensory representation states in the model can be found in the sensory and (pre)motor cortices, indicated in ([20], p. 505) to be 'robustly connected' to the components in the AMPHAC network. One of the roles of the Hippocampus is to store and maintain the relations between sensory memory elements and their emotional associations; in the model these connections are assumed to be fixed and modelled by the (bidirectional) connections between the sensory representations states srs_{sk} and preparation states ps_b of the emotional response b. The feeling state fs_b in the model can be related to the Amygdala, possibly in combination with some limbic areas involved in maintaining 'body maps'. As discussed in Section 2, the interaction between preparation state ps_b and feeling state fs_b is in line with the neurological theories of Damasio [4-7]. About the role of ACC empirical studies show evidence in different directions (e.g., [20], pp. 505-512); therefore it is not clear yet what exactly its function is in dreaming and how it can be related to the model presented in Section 2.

Especially the interaction between MPFC and Amygdala in fear extinction during dreaming has been extensively studied; e.g. [4, 5, 8, 20, 24, 25]. In various empirical studies it has been found that lower activity of MPFC correlates to less controlled feeling levels, and, moreover, REM sleep is found to strengthen MPFC activation and reduce feeling levels; see, for example, [11, 15, 20, 30, 32]. This regulating role of MPFC with respect to Amygdala activation makes these two neurological components suitable candidates for biological counterparts of the control state cs_{skb} and the feeling states fs_b in the computational model presented in Section 3. Moreover, the reported finding suggests that fear extinction learning affects activation of MPFC; this is in accordance with the modelling choice that the Hebbian learning was applied to the two upward connections from sensory representation and feeling states to the control state. As before, the connections between the two types of states may be related to the Hippocampus. Note that in the computational model the control states cs_{skb} also have a role in suppressing the activation of the corresponding sensory representation state srs_{sk} which can be justified as being a form of emotion regulation by attentional deployment; cf. [13, 14]; see also Section 2. The episode states es_{sk} and their competition can be justified by referring to the Global Workspace Theory of consciousness (cf. [1]), as explained in Section 3.

References

1. Baars, B.J.: In the theater of consciousness: the workspace of the mind. Oxford University Press, Oxford (1997)
2. Bi, G., Poo, M.: Synaptic modification by correlated activity: Hebb's postulate revisited. Annu. Rev. Neurosci. 24, 139–166 (2001)
3. Becker, W., Fuchs, A.F.: Prediction in the Oculomotor System: Smooth Pursuit During Transient Disappearance of a Visual Target. Exp. Brain Research 57, 562–575 (1985)
4. Damasio, A.R.: Descartes' Error: Emotion, Reason and the Human Brain. Papermac, London (1994)
5. Damasio, A.R.: The Feeling of What Happens. Body and Emotion in the Making of Consciousness. Harcourt Brace, New York (1999)
6. Damasio, A.R.: Looking for Spinoza: Joy, Sorrow, and the Feeling Brain. Vintage Books, London (2003)
7. Damasio, A.R.: Self comes to mind: constructing the conscious brain. Pantheon Books, NY (2010)
8. Davidson, R.J.: Anxiety and affective style: role of prefrontal cortex and amygdala. Biol. Psychiatry 51, 68–80 (2002)
9. Franzen, P.L., Buysse, D.J., Dahl, R.E., Thompson, W., Siegle, G.J.: Sleep deprivation alters pupillary reactivity to emotional stimuli in healthy young adults. Biol. Psychol. 80, 300–305 (2009)
10. Gerstner, W., Kistler, W.M.: Mathematical formulations of Hebbian learning. Biol. Cybern. 87, 404–415 (2002)
11. Goldin, P.R., McRae, K., Ramel, W., Gross, J.J.: The neural bases of emotion regulation: reappraisal and suppression of negative emotion. Biol. Psychiatry 63, 577–586 (2008)
12. Goldman, A.I.: Simulating Minds: The Philosophy, Psychology, and Neuroscience of Mindreading. Oxford Univ. Press, New York (2006)
13. Gross, J.J.: Antecedent- and response-focused emotion regulation: divergent consequences for experience, expression, and physiology. J. of Personality and Social Psych. 74, 224–237 (1998)
14. Gross, J.J.: Handbook of Emotion Regulation. Guilford Press, New York (2007)
15. Gujar, N., McDonald, S.A., Nishida, M., Walker, M.P.: A Role for REM Sleep in Recalibrating the Sensitivity of the Human Brain to Specific Emotions. Cerebral Cortex 21, 115–123 (2011)
16. Hebb, D.: The Organisation of Behavior. Wiley (1949)
17. Hesslow, G.: Will neuroscience explain consciousness? J. Theoret. Biol. 171, 29–39 (1994)
18. Hesslow, G.: Conscious thought as simulation of behaviour and perception. Trends Cogn. Sci. 6, 242–247 (2002)
19. Hobson, J.A.: REM sleep and dreaming: towards a theory of protoconsciousness. Nature Reviews Neuroscience 10, 803–814 (2009)
20. Levin, R., Nielsen, T.A.: Disturbed dreaming, posttraumatic stress disorder, and affect distress: A review and neurocognitive model. Psychological Bulletin 133, 482–528 (2007)
21. Levin, R., Nielsen, T.A.: Nightmares, bad dreams, and emotion dysregulation. A review and new neurocognitive model of dreaming. Curr. Dir. Psychol. Sci. 18, 84–88 (2009)
22. Nielsen, T.A., Stenstrom, P.: What are the memory sources of dreaming? Nature 437, 1286–1289 (2005)
23. Revonsuo, A.: The reinterpretation of dreams: An evolutionary hypothesis of function of dreaming. Behavioral and Brain Sciences 23, 877–901 (2000)

24. Salzman, C.D., Fusi, S.: Emotion, Cognition, and Mental State Representation in Amygdala and Prefrontal Cortex. Annu. Rev. Neurosci. 33, 173–202 (2010)
25. Sotres-Bayon, F., Bush, D.E., LeDoux, J.E.: Emotional perseveration: an update on prefrontal-amygdala interactions in fear extinction. Learn. Mem. 11, 525–535 (2004)
26. Treur, J.: A Computational Agent Model Using Internal Simulation to Generate Emotional Dream Episodes. In: Samsonovich, A.V., et al. (eds.) Proc. of the Second Intern. Conf. on Biologically Inspired Cognitive Architectures, BICA 2011. IOS Press (in press, 2011)
27. Valli, K., Revonsuo, A., Palkas, O., Ismail, K.H., Ali, K.J., Punamaki, R.L.: The threat simulation theory of the evolutionary function of dreaming: evidence from dreams of traumatized children. Conscious Cogn. 14, 188–218 (2005)
28. Valli, K., Revonsuo, A.: The threat simulation theory in light of recent empirical evidence: a review. Am. J. Psychol. 122, 17–38 (2009)
29. Walker, M.P.: The role of sleep in cognition and emotion. Ann. N Y Acad. Sci. 1156, 168–197 (2009)
30. Walker, M.P., van der Helm, E.: Overnight therapy? The role of sleep in emotional brain processing. Psychol. Bull. 135, 731–748 (2009)
31. Yoo, S.S., Gujar, N., Hu, P., Jolesz, F.A., Walker, M.P.: The human emotional brain without sleep – a prefrontal amygdala disconnect. Curr. Biol. 17, R877–R878 (2007)

Simple Models for Synaptic Information Integration

Danke Zhang[1,2], Yuwei Cui[3], Yuanqing Li[1], and Si Wu[2]

[1] School of Automation Science and Engineering,
South China University of Technology, Guangzhou, China
[2] Institute of Neuroscience, Chinese Academy of Sciences, Shanghai, China
[3] Department of Biology, University of Maryland, College Park, MD, USA

Abstract. Neural information processing is extremely complicated. A core challenge in theoretical neuroscience is to develop properly simplified models, which, on one hand, capture the fundamental features of the complex systems, and on the other hand, allow us to pursue analytic treatments. In the present study, we aim to develop simple models for synaptic information integration. We use simple current-based models to approximate the dynamics of conductance-based multi-compartment ones. The nonlinear shunting inhibition is expressed as a product between the contributions of excitatory and inhibitory currents, and its strength depends on the spatial configuration of excitatory and inhibitory inputs, agreeing with the experimental data. We expect that the current study will serve as a building brick for analyzing the dynamics of large-size networks.

Keywords: Synaptic information integration, Shunting inhibition, Conductance-based model, Current-based model.

1 Introduction

Neural information processing is an extremely complicated process. It involves chemical and electrical interactions at many different levels, including synapses, neurons and networks, to name a few. It also requires information acquisition, propagation, storage, and retrieval in neurons, networks and systems. To make things worse, these interactions are often nonlinear and are implemented in complicated infrastructures. Thus, a core challenge in theoretical neuroscience is to develop properly simplified models, which, on one hand, capture the fundamental features of the complex systems, and, on the other hand, allow us to pursue analytic treatments [2].

The focus of the present study is on developing simple models for synaptic information integration. Experimental data has revealed that the integration of excitatory and inhibitory inputs at the soma of a neuron is a nonlinear process [3], in which a nonlinear shunting inhibition is involved. To get an intuitive idea of the shunting effect, let us consider a simple example: a conductance-based

neuron model receiving both excitatory and inhibitory inputs. The dynamics of the membrane potential of the neuron is given by

$$C\dot{v} = -g^L(v - E_L) - g^E(v - E_e) - g^I(v - E_i), \tag{1}$$

where v is the membrane potential of the neuron, C the membrane capacitance, g^L the leaky conductance and E_L the resting potential. g^E and g^I are, respectively, the excitatory and inhibitory conductances, with E_e and E_i the corresponding reversal potentials. In the stationary state, the membrane potential is given by

$$v = E_L + \frac{g^E(E_e - E_L)}{g^L + g^E + g^I} + \frac{g^I(E_i - E_L)}{g^L + g^E + g^I}. \tag{2}$$

We see that the contribution of the inhibitory input has two folds: apart from inducing a negative term (the third one in the right-hand side of equation), it also decreases the contribution of the excitatory input by enlarging the denominator of the second term. Thus, the integration of excitatory and inhibitory inputs at the soma is more than a linear summation.

The real shunting process is more complicated than the above described, and its effect depends on the spatial configuration of the soma and the locations of excitatory and inhibitory inputs on the dendrites. Interestingly, despite the seemingly complex form, Hao et al. found that the impact of shunting inhibition on the magnitude of somatic potential can be well expressed as a simple arithmetic rule, in which the effect of shunting inhibition is represented as a product between the influences of excitatory and inhibitory inputs applied separately [1].

Motivated by this finding, in the present study, we develop simple current-based models to approximate the dynamics of conductance-based multi-compartmental ones, in particular, to develop a simple way representing the effect of nonlinear shunting inhibition. The current-based neuron models, due to their simplicity, are often used in the theoretic study of large-size networks. Hence, we expect that the current study will serve as a building brick for analyzing the impact of shunting inhibition on the dynamics of large-size networks.

2 A Neuron Model with Soma and a Dendrite

We start by considering a simple integrate-and-fire model consisting of only the soma and a single dendrite. The neuron receives an excitatory and an inhibitory inputs from the locations E and I on the dendrite (Fig.1a). We study first the configuration that the inhibitory input is on the route for the excitatory current propagating to the soma, called the on-path configuration. The neuron fires when its somatic membrane potential exceeds a threshold. Here, our interest is on the sub-threshold dynamics of the neuron.

Below the firing threshold, the dynamics of the local membrane potentials at the soma, the dendritic locations I and E, are given by

$$C^S\dot{v}^S = -g^{LS}(v^S - E_L) - g^{C1}(v^S - v^I), \tag{3}$$

$$C^d\dot{v}^I = -g^{LD}(v^I - E_L) - g^{C1}(v^I - v^S) - g^{C2}(v^I - v^E) - g^I(v^I - E_i), \tag{4}$$

$$C^d\dot{v}^E = -g^{LD}(v^E - E_L) - g^{C2}(v^E - v^I) - g^E(v^E - E_e), \tag{5}$$

where v^S, v^I, and v^E denote, respectively, the local potentials at the soma, the dendrite locations I and E. C^S and C^d are the membrane capacitances of the soma and dendrite, respectively. g^{LS} and g^{LD} are the leaky conductances at the soma and dendrite, respectively. g^{C1} is the transfer conductance between the location I and the soma, g^{C2} the transfer conductance between E and I. Since our focus is on the dendritic information integration, hereafter, for simplicity, we consider g^E and g^I to be constants, equivalent to assuming that the dendrite is receiving continuous external inputs. An equivalent circuit describing the sub-threshold dynamics of the neuron is presented in Fig.1b.
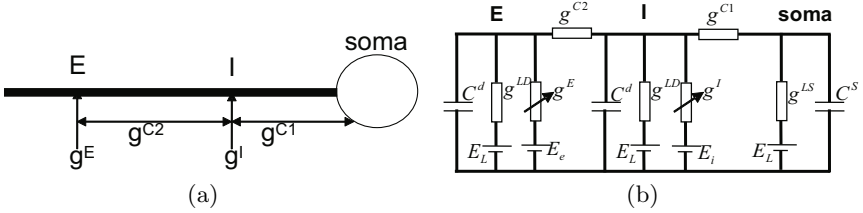

Fig. 1. (a) The spatial configuration of synaptic inputs on the dendrite, with the inhibitory input being on-path from the excitatory one to the soma; (b) The equivalent electrical circuit for describing the sub-threshold dynamics of the neuron

3   A Simplified Current-Based Model

The dynamics given by Eqs.(3-5) is difficult to analyze, in particular, when a network model of many such neurons is concerned. Our goal is to simplify this conductance-based multiple-compartmental model into a simple current-based one, so that they have roughly the same dynamical features.

From Eqs.(3-5), we see that the time constants for the somatic potential, the local potentials at I and E can be roughly estimated to be $\tau_s = C^S/(g^{LS}+g^{C1})$, $\tau_I = C^d/(g^{LD}+g^{C1}+g^{C2})$ and $\tau_E = C^d/(g^{LD}+g^{C2})$. Since the membrane capacitance increases linearly with the surface area of a membrane, we have $C^S \gg C^d$. Moreover, g^{LS} and g^{LD} are in the same order, and $g^{LS} \gg g^{C1}$. Thus, we have $\tau_s \gg \tau_I$ and $\tau_s \gg \tau_E$. This implies that the dynamics of v^I and v^E are much faster than that of v^S. Therefore, we take the following approximation: we consider v^S to be a slow variable, and v^I and v^E reach their steady values at a given v^S instantly.

By setting the left-hand sides of Eqs.(4-5) to be zero, we obtain

$$v^I = E_L + \frac{g^{C1}(v^S - E_L) + g^{C2}(v^E - E_L) + g^I(E_i - E_L)}{g^{LD} + g^{C1} + g^{C2} + g^I}, \tag{6}$$

$$v^E = E_L + \frac{g^{C2}(v^I - E_L) + g^E(E_e - E_L)}{g^{LD} + g^{C2} + g^E}. \tag{7}$$

Substituting the above equations into Eq.(3), we get

$$\tau_s \dot{v}^S = -(v^S - E_L) + f_1(g^E) + f_2(g^I) + k f_1(g^E) f_2(g^I), \tag{8}$$

where

$$f_1(g^E) = \frac{g^{C1}g^{C2}g^E(E_e - E_L)}{g^{LS}(g^L + g^{C1} + g^{LD}/g^{LS}g^{C1})(g^{LD} + g^E)}, \tag{9}$$

$$f_2(g^I) = \frac{g^{C1}g^I(E_i - E_L)}{g^{LS}(g^{LD} + g^{C1} + g^{LD}/g^{LS}g^{C1}) + (g^I + g^{C2})(g^{LS} + g^{C1})}, \tag{10}$$

$$k = \frac{g^{LS} + g^{C1}}{g^{C1}(E_L - E_i)}. \tag{11}$$

To get the above result, we have used the conditions that $g^{C2} \ll g^{LS}$ and $g^{C2} \ll g^{LD}$, provided that E and I are sufficiently separated [4].

It is instructive to look at the steady value of the somatic potential in response to constant synaptic inputs, which is obtained by setting the left-hand side of Eq.(8) to be zero. We get

$$v^S = E_L + f_1(g^E) + f_2(g^I) + k f_1(g^E) f_2(g^I). \tag{12}$$

When no inhibitory input is applied, $f_2(g^I) = 0$, $f_1(g^E) = v^S - E_L$, is the voltage change at the soma due to the excitatory input g^E; when no excitatory input is applied, $f_1(g^E) = 0$, $f_2(g^I) = v^S - E_L$, is the voltage change at the soma due to the inhibitory input g^I. Interestingly, when both excitatory and inhibitory inputs are presented, their joint effect on modifying the somatic potential, i.e, $v^S - E_L$, is expressed as a linear summation of their separated influences, $f_1(g^E)$ and $f_2(g^I)$, with an additional term of their product, i.e., $k f_1(g^E) f_2(g^I)$. This multiplicative term comes from the nonlinear shunting inhibition in the original conductance-based model, and the coefficient k represents the shunting strength. This simple arithmetic rule for the steady value of the somatic potential has been observed in the experiment [1].

The shunting strength is given by Eq.(11), which decreases with the transfer conductance g^{C1}. The latter, on the other hand, is known to decrease with the distance between I and the soma. This tells us that k tends to have a larger value at the distal side of a dendrite than at the proximal side, agreeing with the experimental finding [1].

Similarly, we can compute the case of the out-of-path configuration, where the inhibitory input is not on the route for the excitatory current propagating to the soma. The results are similar, except that $f_1(g^E)$ and $f_2(g^I)$ are replaced by $f_1(g^I)$ and $f_2(g^E)$ accordingly. The new shunting strength is given by $k = (g^{LS} + g^{C1})/\left[g^{C1}(E_l - E_e)\right]$. Since the absolute value of $(E_L - E_e)$ is much smaller than that of $E_L - E_i$ according to the biological data, the shunting strength in this case is much smaller than in the on-path configuration. This agrees with the experimental observation [1].

Fig.2 compares the sub-threshold dynamics of the somatic potential described by the original conductance-based model and the simplified current-based one. In the case of excitatory input dominating (Fig.2a), there is a small discrepancy between the stationary values of two models, but the time costs for them reaching to the steady states are about the same. In the case of inhibitory input dominating (Fig.2b), the time courses of two models agree well. Overall, the simplified model captures the dynamical features of the complex one well.
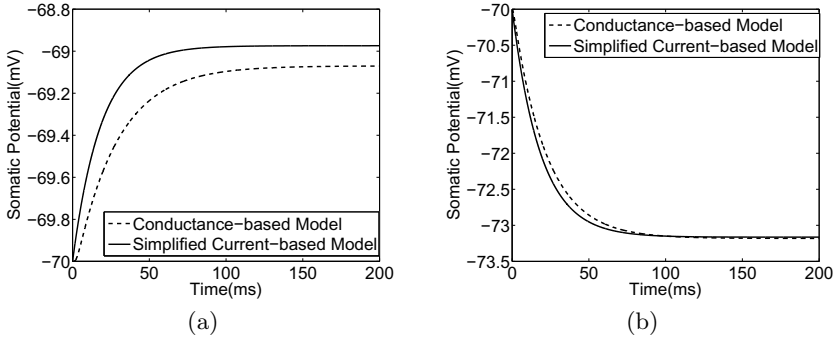

Fig. 2. The conductance-based multi-compartment model vs. the simplified current-based one. (a)The excitatory input dominates, $g^E = 40nS$ and $g^I = 2nS$; (b) The inhibitory input dominates, $g^E = 2nS$ and $g^I = 40nS$. The other parameters are: $C^s = 740pF, C^d = 50pF, g^{LS} = 20nS, g^{LD} = 10nS, g^{C1} = 20nS, g^{C2} = 1nS, E_L = -70mV, E_e = 10mV$ and $E_i = -80mV$.

From Eqs.(9-10), we observe that $f_1(g^E)$ first increases linearly with g^E when $g^E \ll g^{LD}$, and that $f_2(g^I)$ first decreases linearly with g^I when $g^I \ll g^{LS}$. They saturate when g^E and g^I are sufficiently strong (Fig.3). This tells us that for small synaptic inputs, we may approximate $f_1(g^E)$ and $f_2(g^I)$ as linear functions of g^E or g^I, that is, $f_1(g^E) \sim g^E$ and $f_2(g^I) \sim g^I$, and the shunting inhibition is expressed in the form of $hg^E g^I$ with h a proper constant. This linear approximation is very useful when studying the dynamics of a large-size network.

4   Extension to Multiple Dendrites

The above multiplicative rule for representing the shunting inhibition can be extended to more complicated dendrite morphologies. We consider three configurations, which can be related to a number of synaptical connecting patterns in practice.

In the first configuration, individual excitatory or inhibitory inputs are scattered on different dendritic branches (Fig.4a). In this case, the shunting inhibition can be ignored, and the simplified current-based model is written as

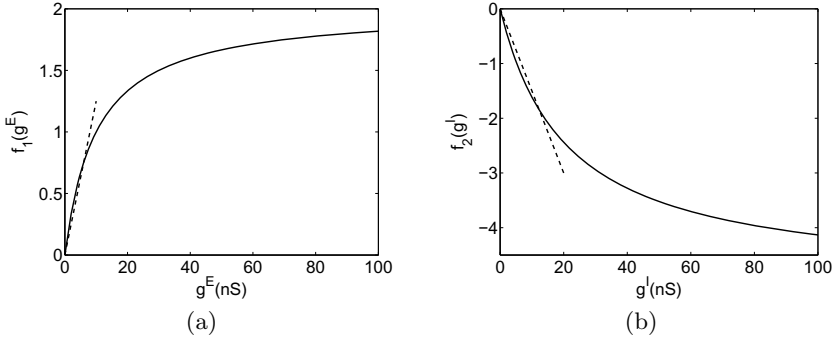$$\tau_s \dot{v}^S = -v^S + E_L + \sum_i f(g_i^{syn}), \tag{13}$$

Fig. 3. (a) The function $f_1(g^E)$; (b) The function $f_2(g^I)$. The dotted lines represent the linear approximation. The parameters are used as in Fig. 2.

where $\tau_s = C^S/(g^{LS} + g^C)$, with g^C the transfer conductance between the input site and the soma. $f(g_i^{syn}) = \sum_i g^C g_i^{syn}(E_i^{syn} - E_L)/(g^{LS} + Ng^C)(g^{LD} + g_i^{syn})$, where N is the number of branches, g_i^{syn} the synaptic conductance on the ith dendritic branch and E_i^{syn} the corresponding reversal potential. g_i^{syn} can be either excitatory or inhibitory.

In the second configuration, pairs of excitatory and inhibitory inputs are scattered on different dendrites (Fig. 4b). In this case, shunting inhibition only exists between a pair on the same dendritic branch. For the on-path configuration, the simplified model is written as

$$\tau_s \dot{v}^S = -v^S + E_L + \sum_i \left[f_1(g_i^E) + f_2(g_i^I) + k f_1(g_i^E) f_2(g_i^I) \right], \qquad (14)$$

where $\tau_s = C^S/(g^L + Ng^C)$, $k = (g^L + Ng^{C1})/\left[g^{C1}(E_L - E_i)\right]$,
$f_2(g_i^I) = g^{C1} g_i^I (E_i - E_L)/\left[(g^{LS} + Ng^{C1})(g^{LD} + g_i^I + g^{C2})\right]$
and $f_1(g_i^E) = g^{C1} g^{C2} g_i^E (E_e - E_L)/\left[g^{LS}(g^{LD} + g_i^E)(g^{LS} + Ng^{C1})\right]$. The result for the out-of-path case can be similarly obtained.

In the third configuration, excitatory inputs are scatted on dendritic branches and an inhibitory input is on the soma or located very close to the soma (Fig. 4c). In this case, the inhibitory input shunts all incoming excitatory currents, termed global shunting, and the simplified model is written as

$$\tau_s \dot{v}^S = -v^S + E_L + \sum_i f_1(g_i^E) + f_2(g^I) + k f_2(g^I) \sum_i f_1(g_i^E), \qquad (15)$$

were $\tau_s = C^S/(g^{LS} + Ng^C + g^I)$, $f_2(g^I) = g^I(E_i - E_L)/(g^{LS} + g^I + Ng^C)$, $f_1(g_i^E) = g^C g_i^E (E_e - E_l)/\left[(g^{LD} + g_i^E)(g^{LS} + Ng^C)\right]$, and $k = 1/(E_L - E_i)$.
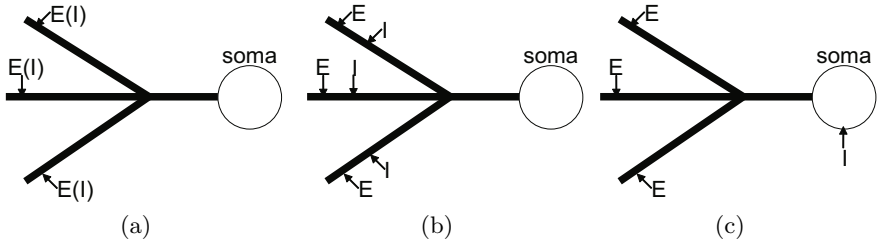
Fig. 4. Three different spatial configurations of synaptic inputs to the soma

5 Conclusions

In the present study, we have derived simple current-based neuron models to approximate the dynamics of conductance-based multi-compartmental ones. In our simplified models, the effect of shunting inhibition is expressed as a product between the contributions of excitatory and inhibitory currents, and the shunting strength depends on the spatial configuration of excitatory and inhibitory inputs, agreeing with the experimental finding [1]. We show that our simplified models can well capture the dynamical features of the original complex ones. The advantage of this approximation is that it simplifies the nonlinear synaptic integration at the single neuron level significantly, and hence provides a foundation for us to simplify the complicated synaptic interaction patterns in large systems. In the future work, we will apply these simplified models to study the dynamics of large-size networks.

References

1. Hao, J., Wang, X., Dan, Y., Poo, M., Zhang, X.: An arithmetic rule for spatial summation of excitatory and inhibitory inputs in pyramidal neurons. Proceedings of the National Academy of Sciences 106, 21906–21911 (2009)
2. Herz, A.V.M., Gollisch, T., Machens, C.K., Jaeger, D.: Modeling Single-Neuron Dynamics and Computations: A Balance of Detail and Abstraction. Science 314, 80–85 (2006)
3. Koch, C., Poggio, T., Torre, V.: Nonlinear Interactions in a Dendritic Tree: Localization, Timing, and Role in Information Processing. Proceedings of the National Academy of Sciences 80, 2799–2802 (1983)
4. Mainen, Z.F., Sejnowski, T.J.: Influence of dendritic structure on firing pattern in model neocortical neurons. Nature 382, 363–366 (1996)

On Rationality of Decision Models Incorporating Emotion-Related Valuing and Hebbian Learning

Jan Treur[1] and Muhammad Umair[1,2]

[1] VU University Amsterdam, Agent Systems Research Group
De Boelelaan 1081, 1081 HV Amsterdam, The Netherlands
{j.treur,m.umair}@vu.nl
[2] COMSATS Institute of Information Technology, Lahore, Pakistan
mumair@ciitlahore.edu.pk
http://www.few.vu.nl/~{treur,mumair}

Abstract. In this paper an adaptive decision model based on predictive loops through feeling states is analysed from the perspective of rationality. Four different variations of Hebbian learning are considered for different types of connections in the decision model. To assess the extent of rationality, a measure is introduced reflecting the environment's behaviour. Simulation results and the extents of rationality of the different models over time are presented and analysed.

Keywords: decision making, cognitive agent model, emotion, Hebbian learning.

1 Introduction

In decision making tasks different options are compared in order to make a reasonable choice out of them. Options usually have emotional responses associated to them relating to a prediction of a rewarding or aversive consequence. In decisions such an emotional valuing often plays an important role. In recent neurological literature this has been related to a notion of value as represented in the amygdala [1, 2, 14, 15, 17]. In making decisions experiences with the environment (from the past) play an important role. By learning processes the decision making mechanism is adapted to these experiences, so that the decision choices made are reasonable or in some way rational, given the enviroment reflected in these past experiences. In this sense the emotion-related valuing in the amygdala as a basis for decision making may be expected to satisfy some rationality criterion. The question to which extent this indeed is the case for certain biologically plausible learning models is the focus of this paper.

The decision model considered involves predictive as-if body loops through feeling states in order to reach decisions for selections of actions (e.g., [3, 6, 8]). The type of learning considered is Hebbian learning (cf. [10, 12]), in four different variations by applying it to different types of connections in the decision model. To assess their extent of rationality, a rationality measure is introduced reflecting the environment's behaviour.

In this paper, in Section 2 the decision model and the different variants of adaptivity considered are introduced. Section 3 presents a number of simulation results. In Section 4 measures for rationality are discussed, and the different models are evaluated. Finally, Section 5 is a discussion.

2   The Adaptive Decision Models Addressed

Traditionally an important function attributed to the amygdala concerns the context of fear. However, in recent years much evidence on the amygdala in humans has been collected showing a function beyond this fear context. In humans many parts of the prefrontal cortex (PFC) and other brain areas such as hippocampus, basal ganglia, and hypothalamus have extensive, often bidirectional connections with the amygdala [11, 15, 18]. A role of amygdala activation has been found in various tasks involving emotional aspects [16]. Usually emotional responses are triggered by stimuli for which a prediction is possible of a rewarding or aversive consequence. Feeling these emotions represents a way of experiencing the value of such a prediction: to which extent it is positive or negative. This idea of value is also the basis of work on the neural basis of economic choice in neuroeconomics. In particular, in decision-making tasks where different options are compared, choices have been related to a notion of value as represented in the amygdala [1, 2, 14, 15, 17, 19].

   Any mental state in a person induces emotions felt by this person, as described in [7, 8, 9]; e.g., [9], p. 93: '… few if any exceptions of any object or event, actually present or recalled from memory, are ever neutral in emotional terms. Through either innate design or by learning, we react to most, perhaps all, objects with emotions, however weak, and subsequent feelings, however feeble.' More specifically, in this paper it is assumed that responses in relation to a sensory representation state roughly proceed according to the following causal chain for a *body loop* (based on elements from [4, 7, 8]):

   sensory representation → preparation for bodily response → body state modification → sensing body state → sensory representation of body state → induced feeling

In addition, an *as-if body loop* uses a direct causal relation

   preparation for bodily response → sensory representation of body state

as a shortcut in the causal chain; cf. [7]. This can be considered a prediction of the action effect by internal simulation (e.g., [13]). The resulting induced feeling is a valuation of this prediction. If the level of the feeling (which is assumed positive) is high, a positive valuation is obtained.

   The body loop (or as-if body loop) is extended to a recursive (as-if) body loop by assuming that the preparation of the bodily response is also affected by the level of the induced feeling:

   induced feeling → preparation for the bodily response

Such recursion is suggested in [8], pp. 91-92, noticing that what is felt is a body state which is under control of the person: 'The brain has a direct means to respond to the object as feelings unfold because the object at the origin is inside the body, rather than external to it. The brain can act directly on the very object it perceives. (…) The object at the origin on the one hand, and the brain map of that object on the other, can influence each other in a sort of reverberative process that is not to be found, for example, in the perception of an external object.' In this way the valuation of the prediction affects the preparation. A high valuation will strengthen activation of the preparation.

Informally described theories in scientific disciplines, for example, in biological or neurological contexts, often are formulated in terms of causal relationships or in terms of dynamical systems. To adequately formalise such a theory the hybrid dynamic modelling language LEADSTO has been developed that subsumes qualitative and quantitative causal relationships, and dynamical systems; cf. [4]. This language has been proven successful in a number of contexts, varying from biochemical processes that make up the dynamics of cell behaviour to neurological and cognitive processes; e.g. [4, 5]. Within LEADSTO a *dynamic property* or temporal relation a $\rightarrow\!\!\!\rightarrow_D$ b denotes that when a state property a occurs, then after a certain time delay (which for each relation instance can be specified as any positive real number D), state property b will occur. Below, this D is the time step Δt. A dedicated software environment is available to support specification and simulation. A specification of the model in LEADSTO format can be found in Appendix A.

An overview of the basic decision model involving the generation of emotional responses and feelings is depicted in Fig. 1. This picture also shows representations from the detailed specifications explained below. However, note that the precise numerical relations are not expressed in this picture, but in the detailed specifications below, through local properties LP0 to LP6.
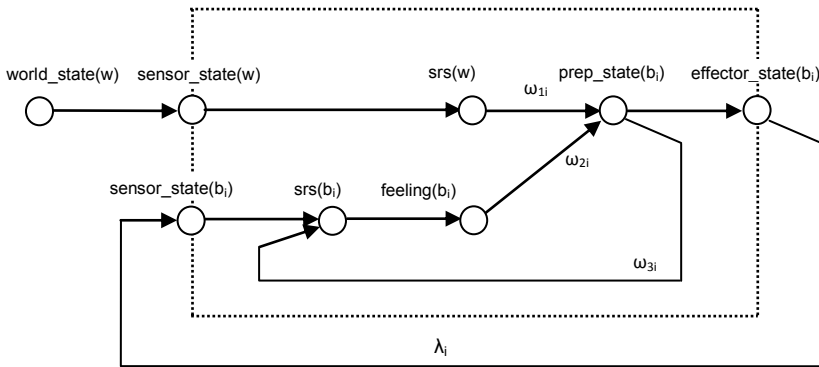

Fig. 1. Overview of the model for decision making evaluated from a rationality perspective

Note that the effector state for b_i combined with the (stochastic) effectiveness of executing b_i in the world (indicated by *effectiveness rate* λ_i between *0* and *1*) activates the sensor state for b_i via body loop as described above. By a recursive as-if body loop each of the preparations for b_i generates a level of feeling for b_i which is considered a valuation of the prediction of the action effect by internal simulation. This in turn affects the level of the related action preparation for b_i. Dynamic interaction within these loops results in equilibrium for the strength of the preparation and of the feeling, and depending on these values, the action is actually activated with a certain intensity. The specific strengths of the connections from the sensory representation to the preparations, and within the recursive as-if body loops can be innate, or are acquired during lifetime. The computational model is based on such neurological notions as valuing in relation to feeling, body loop and as-if body loop. The adaptivity in the

model is based on Hebbian learning. The detailed specification of the model is presented below starting with how the world state is sensed.

LP0 Sensing a World State

If   world state property w occurs of level V_1
  and the sensor state for w occurs has level V_2
then   the sensor state for w will have level $V_2 + \gamma[V_1 - V_2]\,\Delta t$.

$$\frac{dsensor\_state(w)}{dt} = \gamma[world\_state(w) \ - \ sensor\_state(w)] \tag{1}$$

From the sensor state, sensory representation is updated by dynamic property LP1.

LP1 Generating a Sensory Representation for a Sensed World State

If   the sensor state for world state property w has level V_1,
  and the sensory representation for w has level V_2
then   the sensory representation for w will have level $V_2 + \gamma[V_1 - V_2]\,\Delta t$.

$$\frac{dsrs(w)}{dt} = \gamma[sensor\_state(w) \ - \ srs(w)] \tag{2}$$

The combination function h to combine two inputs which activate a subsequent state uses the threshold function th thus keeping the resultant value in the range $[0, 1]$:

$$th(\sigma, \tau, V) = \left(\frac{1}{1+e^{-\sigma(V-\tau)}} - \frac{1}{1+e^{\sigma\tau}}\right) * (1 + e^{-\sigma\tau}) \tag{3}$$

where σ is the steepness and τ is the threshold value. The combination function is:

$$h(\sigma, \tau, V_1, V_2, \omega_1, \omega_2) = \ th\,(\sigma, \tau, \omega_1 V_1 + \omega_2 V_2) \tag{4}$$

where V_1 and V_2 are the current activation level of the states and ω_1 and ω_2 are the connection strength of the links from these states.

    Dynamic property LP2 describes the update of the preparation state for b_i from the sensory representation of w and feeling of b_i.

LP2 From Sensory Representation and Feeling to Preparation of a Body State

If   a sensory representation for w with level V occurs
  and the feeling associated with body state b_i has level V_i
  and the preparation state for b_i has level U_i
  and ω_{1i} is the strength of the connection from sensory representation for w to preparation for b_i
  and ω_{2i} is the strength of the connection from feeling of b_i to preparation for b_i
  and σ_i is the steepness value for preparation of b_i and τ_i is the threshold value for preparation of b_i
  and γ_i is the person's flexibility for bodily responses
then   after Δt the preparation state for body state b_i will have level $U_i + \gamma_i\,[h(\sigma_i, \tau_i, V, V_i, \omega_{1i}, \omega_{2i}) - U_i]\,\Delta t$.

$$\frac{dpreparation(b_i)}{dt} = \gamma[\ h(\sigma_i, \tau_i, srs(w), feeling(b_i), \omega_{1i}, \omega_{2i}) \ - \ preparation(b_i)\ ] \tag{5}$$

Dynamic property LP3 describes the update of the sensory representation of a body state from the respective preparation state and sensor state.

LP3 From Preparation and Sensor State to Sensory Representation of a Body State

If   preparation state for b_i has level X_i
  and sensor state for b_i has level V_i
  and the sensory representation for body state b_i has level U_i

and ω_{3i} is the strength of the connection from preparation state for b_i to sensory representation for b_i
and σ_i is the steepness value for sensory representation of b_i
and τ_i is the threshold value for sensory representation of b_i
and γ_2 is the person's flexibility for bodily responses
then   after Δt the sensory representation for b_i will have level $U_i + \gamma_2 [h(\sigma_i, \tau_i, X_i, V_i, \omega_{3i}, 1)- U_i] \Delta t$.

$$\frac{dsrs(b_i)}{dt} = \gamma[h(\sigma_i, \tau_i, preparation(b_i), sensor\_state(b_i), \omega_3, 1) - srs(b_i)] \tag{6}$$

Dynamic property LP4 describes update of feeling b_i from the sensory representation.

LP4 From Sensory Representation of a Body State to Feeling

If   the sensory representation for body state b_i has level V_1,
   and b_i is felt with level V_2
then b_i will be felt with level $V_2 + \gamma[V_1 - V_2] \Delta t)$.

$$\frac{dfeeling(b_i)}{dt} = \gamma[srs(b_i) - feeling(b_i)] \tag{7}$$

LP5 describes how the effector state for b_i is updated from the preparation state.

LP5 From Preparation to Effector State

If   the preparation state for b_i has level V_1,
   and   the effector state for body state b_i has level V_2.
then   the effector state for body state b_i will have level $V_2 + \gamma[V_1 - V_2] \Delta t$.

$$\frac{deffector\_state(b_i)}{dt} = \gamma[preparation\_state(b_i) - effector\_state(b_i)] \tag{8}$$

LP6 describes update of the sensor state for b_i from the effector state for b_i.

LP6 From Effector State to Sensor State of a Body State

If   the effector state for b_i has level V_1,
   and λ_i is world preference/ recommendation for the option b_i
   and   the sensor state for body state b_i has level V_2,
then   the sensor state for b_i will have level $V_2 + \gamma[\lambda_i V_1 - V_2] \Delta t$

$$\frac{dsensor\_state(b_i)}{dt} = \gamma[\lambda_i effector\_state(b_i) - sensor\_state(b_i)] \tag{9}$$

For the considered case study it was assumed that three options are available to the agent and the objective is to see how rationally an agent makes its decisions using a given adaptive model: under constant as well as in stochastic world characteristics and in both cases static as well as changing worlds. The dynamic properties LP7 to LP9 describe a Hebbian learning mechanism for the connection strengths

   (A) from sensory representation for w to preparation for option b_i
   (B) from feeling b_i to preparation for b_i
   (C) from preparation for b_i to sensory representation of b_i

These have been explored separately (A), (B), or (C), and in combination (ABC).

LP7 Hebbian Learning (A): Connection from Sensory Representation of w to Preparation of b_i

If   the connection from sensory representation of w to preparation of b_i has strength ω_{li}
   and   the sensory representation for w has level V
   and   the preparation of bi has level V_i
   and   the learning rate from sensory representation of w to preparation of b_i is η

and   the extinction rate from sensory representation of w to preparation of b_i is ζ
then   after Δt the connection from sensory representation of w to preparation of bi will have
strength $\omega_{1i} + (\eta VV_i(1 - \omega_{1i}) - \zeta\omega_{1i})\, \Delta t$.

$$\frac{d\omega_{1i}}{dt} = \eta\, srs(w)\, preparation(b_i)\, (1 - \omega_{1i}) - \zeta\omega_{1i} \tag{10}$$

LP8 Hebbian Learning (B): Connection from Feeling b_i to Preparation of b_i

If   the connection from feeling associated with body state b_i to preparation of b_i has strength ω_{2i}
and   the feeling for bi has level V_i
and   the preparation of bi has level U_i
and   the learning rate from feeling of bi to preparation of b_i is η
and   the extinction rate from feeling of bi to preparation of b_i is ζ
then   after Δt the connection from feeling of bi to preparation of bi will have
strength $\omega_{2i} + (\eta V_i U_i(1 - \omega_{2i}) - \zeta\omega_{2i})\, \Delta t$.

$$\frac{d\omega_{2i}}{dt} = \eta\, feeling(b_i) preparation(b_i)\, (1 - \omega_{2i}) - \zeta\omega_{2i} \tag{11}$$

LP9 Hebbian Learning (C): Connection from Preparation of b_i to Sensory Representation of b_i

If   the connection from preparation of b_i to sensory representation of b_i has strength ω_{3i}
and   the preparation of bi has level V_i and   the sensory representation of bi has level U_i
and   the learning rate from preparation of bi to sensory representation of b_i is η
and   the extinction rate from preparation of bi to sensory representation of b_i is ζ
then   after Δt the connection from preparation of bi to sensory representation of bi will have
strength $\omega_{3i} + (\eta V_i U_i(1 - \omega_{3i}) - \zeta\omega_{3i})\, \Delta t$.

$$\frac{d\omega_{3i}}{dt} = \eta\, preparation(b_i)\, srs(b_i)\, (1 - \omega_{3i}) - \zeta\omega_{3i} \tag{12}$$

3   Simulation Results

In this section some of the simulation results, performed using numerical software, are described in detail. The simulation results address different scenarios reflecting different types of world characteristics, from constant to stochastic world, and from static to changing world. Moreover, learning the connections was done one at a time (A), (B), (C), and learning multiple connections simultaneously (ABC). Due to space limitation the graphs for only (A) are shown here. A summary of the results is given in Table 1. Results for the rationality factors are presented in the next section. For all simulation results shown, time is on the horizontal axis whereas the vertical axis shows the activation level of the different states. Step size for all simulations is $\Delta t = 1$. Fig. 2 shows simulation results for the model under constant, static world characteristics: $\lambda_1 = 0.9$, $\lambda_2 = 0.2$, and $\lambda_3 = 0.1$. Other parameters are set as: learning rate $\eta = 0.04$, extinction rate $\zeta = 0.0015$, initial connection strength $\omega_{2i} = \omega_{3i} = 0.8$, speed factors $\gamma = 1$, $\gamma_1 = 0.5$, $\gamma_2 = 1$, steepness $\sigma = 2$ and threshold $\tau = 1.2$ for preparation state, and $\sigma = 10$ and $\tau = 0.3$ for sensory representation of b_i. For initial 80 time units the stimulus w is kept 1 and for next 170 time units it is kept 0 and same sequence of activation and deactivation for stimulus is repeated for rest of simulation.
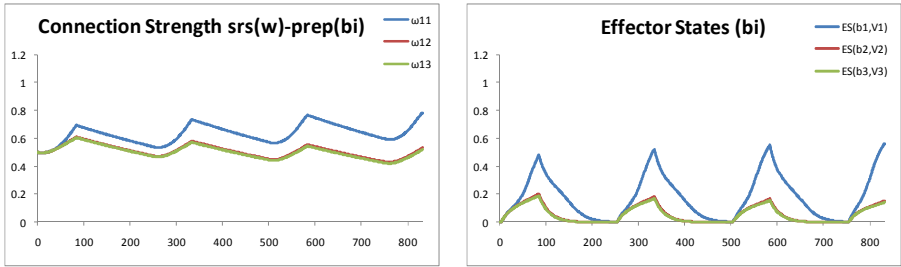
Fig. 2. Constant World: (a) Connection strengths (A)   (b) Effector States for b_i
Initial values $\omega_{11}= \omega_{12}= \omega_{13}= 0.5$; $\eta= 0.04$, $\zeta= 0.0015$

Moreover it depicts the situation in which only one type of links (ω_{1i}) is learned as specified in LP7 using the Hebbian approach (A) for the connection from sensory representation of w to preparation state for b_i. It is shown that the model adapts the connection strengths of the links ω_{1i} according to the world characteristics given by λ_i. So ω_{11} strengthens more and more over time, resulting in the higher activation level of the effector state for b_1 compared to the activation level of the effector states for the other two options b_2 and b_3.

Similar experiments were carried out for a stochastic world with four different cases as mentioned earlier. To simulate the stochastic world, probability distribution functions (PDF) were defined for λ_i according to a Normal Distribution. Using these PDFs, the random numbers were generated for λ_i limiting the values for the interval $[0, 1]$ with $\mu_1=0.9$, $\mu_2=0.2$ and $\mu_3=0.1$ for λ_i respectively. Furthermore the standard deviation for all λ_i was taken 0.1. Fig. 3 shows the world state w and stochastic world characteristics λ_i. Fig. 4 shows the simulation results while learning is performed for the links (A) from sensory representation of w to preparation state for b_i.
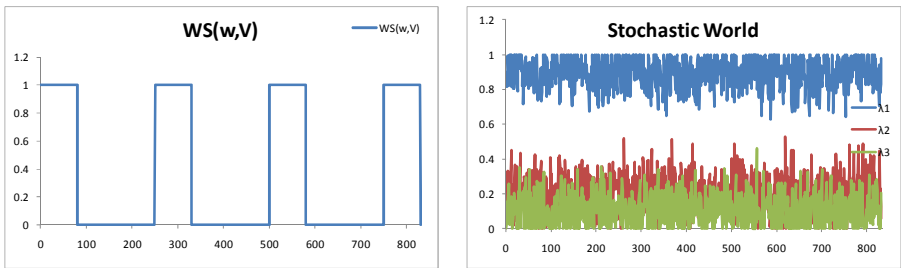

Fig. 3. Stochastic World

It can be seen from these results that also in a stochastic scenario the agent model successfully learnt the connections and adapted to the world characteristics rationally with results quite similar to the results for a static world.
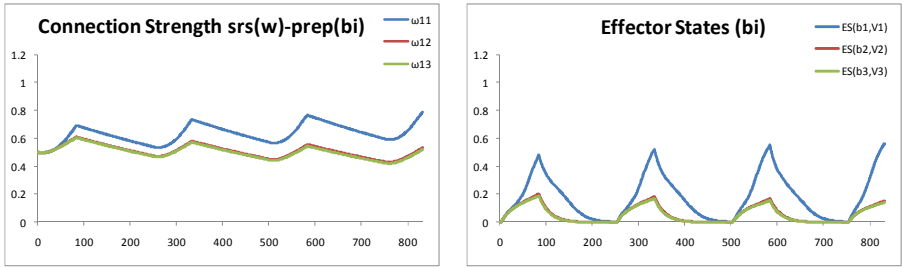
Fig. 4. Stochastic World: (a) Connection strengths (A)   (b) Effector States
Initial values $\omega_{11}= \omega_{12}= \omega_{13}= 0.5$; $\eta= 0.04$, $\zeta= 0.0015$

Another scenario was explored in which the (stochastic) world characteristics were changing drastically from $\mu_1=0.9$, $\mu_2=0.2$ and $\mu_3=0.1$ for λ_i respectively to $\mu_1=0.1$, $\mu_2=0.2$ and $\mu_3=0.9$ for λ_i respectively with standard deviation of 0.1 for all. Fig. 5 and Fig. 6 show the results for such a scenario. The results show that the agent has successfully adapted to the changing world characteristics over time. The initial settings in this experiment were taken from the previous simulation results shown in Figs. 3 and 4 to keep the continuity of the experiment. It can be observed that the connection strength for option 3 becomes higher compared to the other options, and consequently the value of the effector state for b_3 becomes higher than for the other two by the end of experiment.
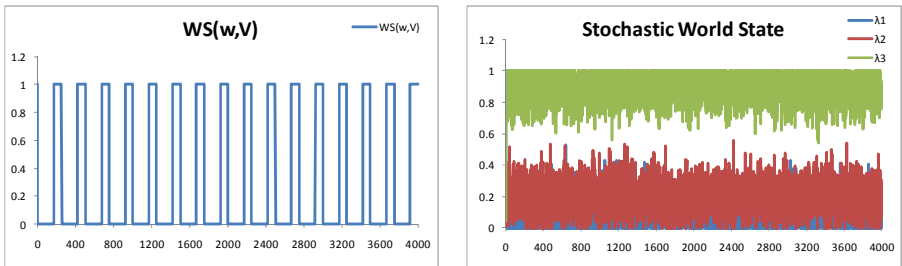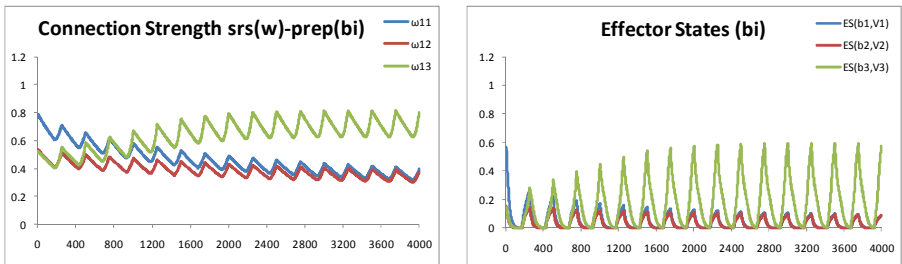

Fig. 5. World State

Fig. 6. Changing World: (a) Connection strengths (A)   (b) Effector States
Initial values $\omega_{11}=0.78$, $\omega_{12}= 0.53$, $\omega_{13}= 0.52$; $\eta= 0.04$, $\zeta= 0.0015$

Table 1. Overview of the simulation results for all links (A), (B), (C) and (ABC)

Link	Scenario	ω_{x1}	ω_{x2}	ω_{x3}	ES_1	ES_2	ES_3
A	Static	0.78	0.53	0.52	0.56	0.15	0.14
	Stocastic	0.78	0.53	0.52	0.56	0.15	0.14
	Change World	0.40	0.38	0.80	0.09	0.09	0.58
B	Static	0.89	0.58	0.46	0.65	0.38	0.31
	Stochastic	0.89	0.59	0.47	0.65	0.39	0.32
	Change World	0.42	0.57	0.89	0.30	0.37	0.65
C	Static	0.88	0.29	0.23	0.63	0.28	0.26
	Stochastic	0.88	0.29	0.23	0.63	0.28	0.27
	Change World	0.04	0.08	0.87	0.25	0.26	0.63
ABC	Static	0.81	0.55	0.54	0.59	0.13	0.13
		0.85	0.30	0.29			
		0.85	0.30	0.29			
	Stocastic	0.80	0.55	0.54	0.57	0.13	0.13
		0.84	0.30	0.29			
		0.84	0.30	0.29			
	Changed World	0.64	0.64	0.94	0.16	0.16	0.75
		0.02	0.03	0.96			
		0.02	0.03	0.96			

Similar results were observed for all other cases (B), (C), and (ABC) as summarised in Table 1. Note that the table contains the values of different connection strengths and the activation level of effector states after the completion of simulation experiments. This shows a rational behavior of the agent in this particular scenario.

4   Evaluating Agent Models on Rationality

In the previous section it was shown that the agent model behaves rationally in different scenarios. These scenarios and its different cases are elaborated in detail in the previous section, but the results were assessed with respect to their rationality in a qualitative and rather informal manner. For example, no attempt was made to assign an extent or level to the rationality observed during these experiments. The current section addresses this and to this end two different formally defined measures to assess the extent of the rationality are introduced; one rationality measure is based on a discrete scale and the other one on a continuous scale.

Method 1 (Discrete Rationality Measure)

The first method presented is based on the following point of departure: *an agent which has the same respective order of effector state activation levels for the different options compared to the order of world characteristics λ_i will be considered highly rational*. So in this method the rank of the average value λ_i at any given time unit is

determined, and compared with the rank of the respective effector state levels. More specifically, the following formula is used to determine the irrationality factor IF.

$$IF = \sum_{i=1}^{n} abs(rank(es_i) - rank(\lambda_i)) \tag{13}$$

where n is the number of options available. This irrationality factor tells to which extent the agent is behaving rationally in the sense that the higher the irrationality factor IF is, the lower is the rationality of the agent. It is assumed that the there is uniqueness in ranking and none of the two values assign a similar rank. To calculate the discrete rationality factor DRF, the maximum possible irrationality factor $Max.\ IF$ can be determined as follows.

$$Max.\ IF = \frac{n(n+1)}{2} - ceiling\left(\frac{n}{2}\right) \tag{14}$$

Here $ceiling(x)$ is the first integer higher than x. Note that $Max.\ IF$ is approximately $\frac{1}{2}n^2$. As a higher IF means lower rationality, the discrete rationality factor DRF is calculated as:

$$DRF = 1 - \frac{IF}{Max.\ IF} \tag{15}$$

On this scale, for each n only a limited number of values are possible; for example, for $n = 3$ three values are possible: 0, 0.5, and 1. In general $\frac{1}{2}\ Max.\ IF\ +1$ values are possible, which is approximately $\frac{1}{4}n^2 + 1$. As an example, suppose during a simulation average values of $\lambda_1 = 0.107636$, $\lambda_2 = 0.203044$, and $\lambda_3 = 0.888522$ are given, whereas the effector state values are $ES_1 = 0.170554$, $ES_2 = 0.12367$ and $ES_3 = 0.43477$ at a given time point. So according to the given data the world's ranks will be 3, 2, 1 for λ_1, λ_2, λ_3 and the agent's ranks 2, 3, 1 for ES_1, ES_2, ES_3 respectively. So according to the given formulas $IF = 2$, $Max.\ IF = 4$ and $DRF = 0.5$. So in this particular case at this given time point the agent is behaving rationally for 50%.

Method 2 (Continuous Rationality Measure)

The second method presented is based on the following point of departure: *an agent which receives the maximum benefit will be the highly rational agent. This is only possible if ES_i is 1 for the option whose λ_i is the highest*. In this method to calculate the continuous rationality factor CRF, first to account for the effort spent in performing actions, the effector state values ES_i are normalised as follows.

$$nES_i = \frac{ES_i}{\sum_{i=1}^{n} ES_i} \tag{16}$$

Here n is number of options available. Based on this the continuous rationality factor CRF is determined as follows, with $Max(\lambda_i)$ the maximal value of the different λ_i.

$$CRF = \frac{\sum_{i=1}^{n} nES_i\,\lambda_i}{Max(\lambda_i)} \tag{17}$$

This method enables to measure to which extent the agent is behaving rationally in a continuous manner. For the given example used to illustrate the previous method $CRF = 0.6633$. So according to this method the agent is considered to behaving for 66.33% rationally in the given world.
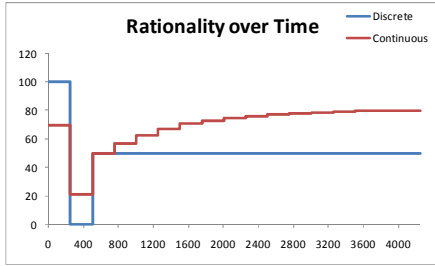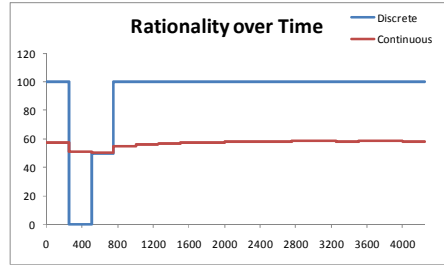
Fig. 7. Rationality during learning ω_{1i} (A)
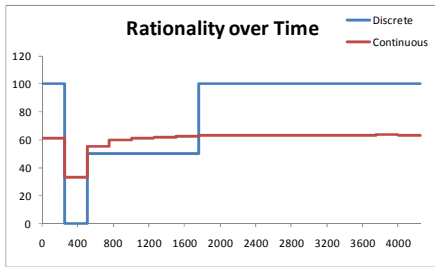

Fig. 8. Rationality during learning ω_{2i} (B)

Fig. 9. Rationality during learning ω_{3i} (C)
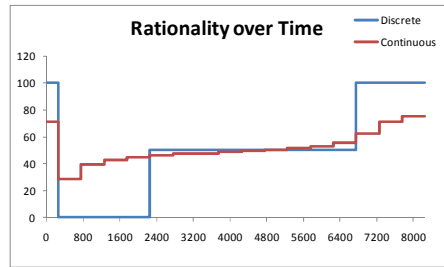
Fig. 10. Rationality for learning ω_{1i}, ω_{2i}, ω_{3i} (ABC)

Fig. 7 to Fig. 10 show the two types of rationality (depicted as percentages) of the agent for the different scenarios with changing stochastic world. In these figures the first 250 time points show the rationality achieved by the agent just before changing world characteristics drastically for the simulations shown from Fig. 4. From time point 250 onwards, it shows the rationality of the agent after the change has been made (see Fig. 6). It is clear from the results (Fig. 7 to Fig. 10) that the rationality factor of the agent in all four cases improves over time for the given world.

5   Discussion

This paper focused on how the extent of rationality of an adaptive decision model can be analysed. In particular, this was explored for variants of a decision model based on valuing of predictions involving feeling states generated in the amygdala; e.g., [1, 2, 6, 8, 14, 15, 17]. The adaptation was based on using four different variations of Hebbian learning; cf. [10, 12].

To assess the extent of rationality with respect to given world characteristics, two measures were introduced, and using these extents of rationality of the different models over time were analysed. It was shown how by the learning processes indeed a high level of rationality was obtained, and how after a major world change after some delay this rationality level is re-obtained. It turned out that emotion-related valuing of predictions in the amygdala as a basis for adaptive decision making according to Hebbian learning satisfies reasonable rationality measures.

References

1. Bechara, A., Damasio, H., Damasio, A.R.: Role of the Amygdala in Decision-Making. Ann. N.Y. Acad. Sci. 985, 356–369 (2003)
2. Bechara, A., Damasio, H., Damasio, A.R., Lee, G.P.: Different Contributions of the Human Amygdala and Ventromedial Prefrontal Cortex to Decision-Making. Journal of Neuroscience 19, 5473–5481 (1999)
3. Bosse, T., Hoogendoorn, M., Memon, Z.A., Treur, J., Umair, M.: An Adaptive Model for Dynamics of Desiring and Feeling Based on Hebbian Learning. In: Yao, Y., Sun, R., Poggio, T., Liu, J., Zhong, N., Huang, J. (eds.) BI 2010. LNCS, vol. 6334, pp. 14–28. Springer, Heidelberg (2010)
4. Bosse, T., Jonker, C.M., Treur, J.: Formalisation of Damasio's Theory of Emotion, Feeling and Core Consciousness. Consciousness and Cogn. 17, 94–113 (2008)
5. Bosse, T., Jonker, C.M., van der Meij, L., Treur, J.: A Language and Environment for Analysis of Dynamics by Simulation. Intern. J. of AI Tools 16, 435–464 (2007)
6. Damasio, A.: Descartes' Error: Emotion, Reason and the Human Brain. Papermac, London (1994)
7. Damasio, A.: The Feeling of What Happens. Body and Emotion in the Making of Consciousness. Harcourt Brace, New York (1999)
8. Damasio, A.: Looking for Spinoza. Vintage Books, London (2004)
9. Forgas, J.P., Laham, S.M., Vargas, P.T.: Mood effects on eyewitness memory: Affective influences on susceptibility to misinformation. Journal of Experimental Social Psychology 41, 574–588 (2005)
10. Gerstner, W., Kistler, W.M.: Mathematical formulations of Hebbian learning. Biol. Cybern. 87, 404–415 (2002)
11. Ghashghaei, H.T., Hilgetag, C.C., Barbas, H.: Sequence of information processing for emotions based on the anatomic dialogue between prefrontal cortex and amygdala. Neuroimage 34, 905–923 (2007)
12. Hebb, D.O.: The Organization of Behaviour. John Wiley & Sons, New York (1949)
13. Hesslow, G.: Conscious thought as simulation of behaviour and perception. Trends Cogn. Sci. 6, 242–247 (2002)
14. Montague, P.R., Berns, G.S.: Neural economics and the biological substrates of valuation. Neuron. 36, 265–284 (2002)
15. Morrison, S.E., Salzman, C.D.: Re-valuing the amygdala. Current Opinion in Neurobiology 20, 221–230 (2010)
16. Murray, E.A.: The amygdala, reward and emotion. Trends Cogn. Sci. 11, 489–497 (2007)
17. Rangel, A., Camerer, C., Montague, P.R.: A framework for studying the neurobiology of value-based decision making. Nat. Rev. Neurosci. 9, 545–556 (2008)
18. Salzman, C.D., Fusi, S.: Emotion, Cognition, and Mental State Representation in Amygdala and Prefrontal Cortex. Annu. Rev. Neurosci. 33, 173–202 (2010)
19. Sugrue, L.P., Corrado, G.S., Newsome, W.T.: Choosing the greater of two goods: neural currencies for valuation and decision making. Nat. Rev. Neurosci. 6, 363–375 (2005)

Appendix A Model Specifications in LEADSTO Format

LP0 Sensing a world state
world_state(w, V1) & sensor_state(w, V2) $\rightarrow$ sensor_state(w, V2 + γ [V1 – V2] Δt)

LP1 Generating a sensory representation for a sensed world state
sensor_state(w, V1) & srs(w, V2) $\rightarrow$ srs(w, V2 + γ [V1 – V2] Δt)

LP2 From sensory representation and feeling to preparation of a body state
srs(w, V) & feeling(b_i, V_i) & preparation_state(b_i, U_i) &
has_connection_strength(srs(w), preparation(bi), ω_{1i}) &
has_connection_strength(feeling(b_i), preparation(b_i), ω_{2i}) &
has_steepness(prep_state(b_i), σ_i) & has_threshold(prep_state(b_i), τ_i)
$\rightarrow$ preparation(b_i, U_i + γ_1 (h(σ_i, τ_i, V, V_i, ω_{1i}, ω_{2i}) - U_i) Δt)

LP3 From preparation and sensor state to sensory representation of a body state
preparation_state (b_i, X_i) & sensor_state(b_i, V_i) & srs (b_i, U_i) &
has_connection_strength(preparation(bi), srs(bi), ω_{3i}) & has_steepness(srs(b_i), σ_i) &
has_threshold(srs(b_i), τ_i) $\rightarrow$ srs(b_i, U_i + γ_2 (h(σ_i, τ_i, X_i, V_i, ω_{3i}, 1) - U_i) Δt)

LP4 From sensory representation of a body state to feeling
srs(bi, V1) & feeling(bi, V2) $\rightarrow$ feeling(bi, V2 + γ [V1 – V2] Δt)

LP5 From preparation to effector state
preparation_state(bi, V) & effector_state(bi, V2) $\rightarrow$ effector_state(bi, V2 + γ [V1 – V2] Δt))

LP6 From effector state to sensor state of a body state
effector_state(bi,V1) & effectiveness_rate(bi, λ_i) & sensor_state(bi,V2)
$\rightarrow$ sensor_state(bi, V2 + γ [λ_i V1 – V2] Δt)

LP7 Hebbian learning (A): connection from sensory representation of *w* to preparation of b_i
has_connection_strength(srs(w), preparation(b_i), ω_{1i}) & srs(w, V) & preparation(b_i, V_i) &
has_learning_rate(srs(w), preparation(b_i), η) & has_extinction_rate(srs(w), preparation(b_i), ζ)
$\rightarrow$ has_connection_strength(w, b_i, ω_{1i} + (ηVV$_i$ (1 - ω_{1i}) - $\zeta\omega_{1i}$) Δt)

LP8 Hebbian learning (B): connection from feeling b_i to preparation of b_i
has_connection_strength(feeling(bi), preparation(bi), ω_{2i}) & feeling(b_i, V_i) & preparation(b_i, U_i) &
has_learning_rate(feeling(b_i), preparation(b_i), η) & has_extinction_rate(feeling(b_i), preparation(b_i), ζ)
$\rightarrow$ has_connection_strength(feeling(bi), preparation(bi), ω_{2i} + (ηV$_i$U$_i$ (1 - ω_{2i}) - $\zeta\omega_{2i}$) Δt)

LP9 Hebbian learning (C): connection from preparation of b_i to sensory representation of b_i
has_connection_strength(preparation (bi), srs(b_i), ω_{3i}) & preparation (b_i, V_i) & srs(b_i, U_i) &
has_learning_rate(preparation (b_i), srs(b_i), η) & has_extinction_rate(preparation (b_i), srs(b_i), ζ)
$\rightarrow$ has_connection_strength(preparation (bi), srs(b_i), ω_{3i} + (ηV$_i$U$_i$ (1 - ω_{3i}) - $\zeta\omega_{3i}$) Δt)

Evolving Probabilistic Spiking Neural Networks for Spatio-temporal Pattern Recognition: A Preliminary Study on Moving Object Recognition

Nikola Kasabov[1,2], Kshitij Dhoble[1], Nuttapod Nuntalid[1], and Ammar Mohemmed[1]

[1] Knowledge Engineering and Discovery Research Institute,
Auckland University of Technology,
Private Bag 92006, Auckland 1010, New Zealand
{nkasabov,kdhoble,nnuntali,amohemme}@aut.ac.nz
http://www.kedri.info
[2] Institute for Neuroinformatics,
University of Zurich and ETH Zurich

Abstract. This paper proposes a novel architecture for continuous spatio-temporal data modeling and pattern recognition utilizing evolving probabilistic spiking neural network 'reservoirs' (epSNNr). The paper demonstrates on a simple experimental data for moving object recognition that: (1) The epSNNr approach is more accurate and flexible than using standard SNN; (2) The use of probabilistic neuronal models is superior in several aspects when compared with the traditional deterministic SNN models, including a better performance on noisy data.

Keywords: Spatio-Temporal Patterns, Spiking Neural Network, Reservoir Computing, Liquid State Machine.

1 Introduction

Video information is spatio-temporal (ST) in nature and the problem of ST pattern recognition (STPR) is a challenging task in the machine learning domain. Existing statistical and artificial neural networks machine learning approaches fail to model the complex ST dynamics optimally, since they either process spatial and temporal component separately or integrate them together in a simple way, losing the significant correlation information present in the ST data. Many of the existing methods process data on a frame-by-frame bases, rather than as whole spatio-temporal patterns.

Hidden Markov Models (HMM) is among the most popular statistical approaches, widely used for processing time series [1]. HMM are often used either with traditional neural networks [2] or on its own [3]. However, HMM have some limitations when used for multiple times series that have spatial components too [4].

There are other emerging approaches such as deep machine learning which involves the combination of Deep Belief Networks (DBNs - Generative Model) and Convolutional Neural Networks (CNNs - Discriminative Model) [5]. The proposed DBNs model nevertheless carries out learning in a frame by frame manner, rather than learning the entire STD patterns.

The brain inspired SNN have the ability to learn spatio-temporal patterns by using trains of spikes (which are spatiotemporal events) [6]. Furthermore, the 3D topology of a spiking neural network reservoir has the potential to capture a whole STD pattern at any given time point. The neurons in this reservoir system transmit spikes via synapses that are dynamic in nature, collectively forming a ST memory [7]. Often, learning rules such as Spike-Time-Dependent-Plasticity (STDP) [8] are commonly utilized in SNN models.

Recently, several SNN models and their applications have been developed by numerous research groups [9],[10] as well as by our research group [11], [12], [13]. However, they still process ST data as a sequence of static feature vectors extracted from segments of data, without utilizing the SNN's capability of learning whole ST patterns.

In order to address the limitations of the current machine learning techniques for ST pattern recognition from continuous ST data, we have developed a novel SNN architecture called evolving probabilistic SNN reservoir (epSNNr).

2   The Proposed epSNNr Architecture

The proposed epSNNr architecture is characterised in principle by the following characteristics:

- its structure is evolving from input data;
- it uses a probabilistic model of a neuron;
- it captures in its internal space ST patterns from data that can be classified in an output module;

The design of the overall pSNNr architecture is illustrated in Fig. 1, where the data acquisition part represents the video and/or audio data stream along with the spike encoding module. The data processing module represents several components/modules where dimensional transformation and learning takes place.

The connections between neurons are initially set using a Gaussian function centered at each spatially located neuron, so that closer neurons are connected with a higher probability. The input information is transformed into trains of spikes before being submitted to the epSNNr. Continuous value input variables can be transformed into spikes using different approaches:

- population rank coding [14],[11],[12];
- thresholding the input value, so that a spike is generated if the input value is above a threshold;
- thresholding the difference between two consecutive values of the same variable over time as it is in the artificial cochlea and artificial retina devices [15],[16].
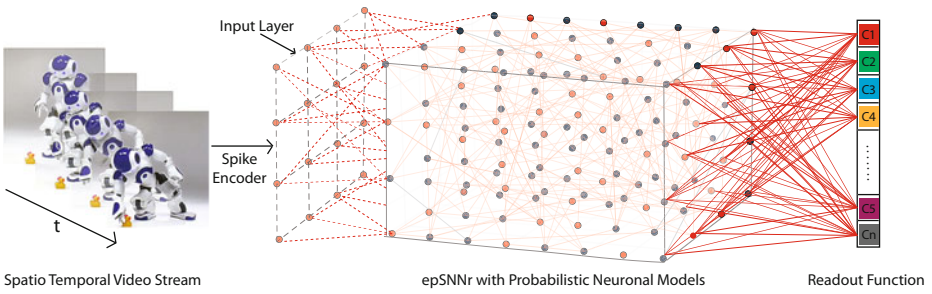
Fig. 1. A generic epSNNr architecture for ST data modeling and pattern recognition

The input information is entered in the epSNNr continuously and its state is evaluated after a 'chunk' of the input stream is entered, rather than after every single time frame.

The epSNNr uses a probabilistic neural model as explained in the next section. The current state of the epSNN 'reservoir' $S(t)$ is captured in an output module. For this purpose dynamically created spatio-temporal clusters $C_1, C_2, \ldots C_k$ of close (both in space and time) neurons, can be used. The state of each cluster C_i at a time t is represented by a single number, reflecting on the spiking activity at this time moment of all neurons in the cluster, which is interpreted as the current spiking probability of the cluster. The states of all clusters define the current reservoir state $S(t)$. In the output function, the cluster states are used differently for different tasks.

3   Probabilistic Neuronal Models in the epSNNr as Extensions of the LIF Model

Models of probabilistic neurons have been proposed in several studies, e.g. in the form of dynamic synapses [16], the stochastic integration of the post-synaptic potential [17] and stochastic firing thresholds [18]. In [13] a probabilistic neuronal model is introduced that has three probabilistic parameters to extend the LIF model:

- $p_{cj,i}(t)$ is the probability that a spike emitted by neuron n_j will reach neuron n_i at a time moment t trough the connection between n_j and n_i ;
- $p_{sj,i}(t)$ is the probability of the synapse $s_{j,i}$ to contribute to the post synaptic potential $PSPi(t)$ after the latter has received a spike from neuron n_j;
- $p_i(t)$ is the probability parameter for the neuron n_i to emit an output spike at time t, once the total post-synaptic potential $PSPi(t)$ has reached a value above the PSP threshold (a noisy threshold).

As a partial case, when all or some of the probability parameters are fixed to "1", the pSNM can be reduced to the LIF. The LIF neuron is arguably the best known model for simulating spiking networks. It is based on the idea of

an electrical circuit containing a capacitor with capacitance C and a resistor with resistance R, where both C and R are assumed to be constant. The model dynamics are described by the following differential equation:

$$\tau_m \frac{du}{dt} = -u(t) + RI(t) \tag{1}$$

The constant τ_m is called the membrane time constant of the neuron. Whenever the membrane potential u crosses a threshold v from below, the neuron fires a spike and its potential is reset to a resting potential u_r. It is noteworthy that the shape of the spike itself is not explicitly described in the traditional LIF model. Only the firing times are considered to be relevant. We will introduce here only three types of probabilistic models considering only the third probability parameter $p_i(t)$ of the probabilistic model from [13]. The rest of the probability parameters are not considered in this study or assumed to be set to 1.

We define a *stochastic reset* (SR) model that replaces the deterministic reset of the potential after spike generation with a stochastic one. Let $t^{(f)} : u(t^{(f)}) = v$ be the firing time of a LIF neuron, then

$$\lim_{t \to t^{(f)}, t > t^{(f)}} u(t) = N(u_r, \sigma_{SR}) \tag{2}$$

defines the reset of the post-synaptic potential. $N(u_r, \sigma_{SR})$ is a Gaussian distributed random variable with mean μ and standard deviation σ. Variable σ_{ST} represents a parameter of the model.

We define two stochastic threshold models that replace the constant firing threshold v of the LIF model with a stochastic one. In the *step-wise stochastic threshold* (ST) model, the dynamics of the threshold update are defined as

$$\lim_{t \to t^{(f)}, t > t^{(f)}} v(t) = N(v_0, \sigma_{ST}) \tag{3}$$

Variable σ_{ST} represents the standard deviation of the Gaussian distribution N and is a parameter of the model. According to Eq.2, the threshold is the outcome of a v_0-centered Gaussian random variable which is sampled whenever the neuron fires. We note that this model does not allow spontaneous spike activity. More specifically, the neuron can only spike at time $t^{(f)}$ when also receiving a pre-synaptic input spike at $t^{(f)}$. Without such a stimulus a spike output is not possible. The *continuous stochastic threshold* (CT) model updates the threshold continuously over time. Consequently, this model allows spontaneous spike activity, i.e, a neuron may spike at time even in the absence of a pre-synaptic input spike . The threshold is defined as an Ornstein-Uhlenbeck process [19]:

$$\tau_v \frac{dv}{dt} = v_0 - v(t) + \sigma_{CT} \sqrt{2\tau_v \xi(t)} \tag{4}$$

where the noise term ξ corresponds to Gaussian white noise with zero mean and unit standard deviation. Variable σ_{CT} represents the standard deviation of the fluctuations of $v(t)$ and is a parameter of the model. We note that $v(t)$ has an

overall drift to a mean value v_0 , i.e, $v(t)$ reverts to v_0 exponentially with rate τ_v, the magnitude being in direct proportion to the distance $v_0 - v(t)$.

In this paper we explore the feasibility of using the above three probabilistic models in an epSNNr for a simple moving object recognition task.

4   Preliminary Experiments on Moving Object Recognition in the epSNNr

4.1   Goals of the Experimental Study

The aim of this study is to demonstrates the feasibility of the proposed a novel architecture for continuous ST modeling and pattern recognition utilizing epSNNr. More specifically, in this study we show that: (1) The epSNNr approach is more accurate and flexible than using standard SNN; (2) The use of probabilistic neuronal models is superior when compared with the traditional deterministic SNN models, including a better performance on noisy data.

In order to demonstrate the feasibility of the proposed novel architecture, we have evaluated our approach on a synthetic video dataset that has been described in the following subsection.

4.2   Synthetic Video Dataset

The synthetic video data set (see Fig. 2) consists of 4 different classes having 5 samples in each class. Each class corresponds to the objects trajectory / movement
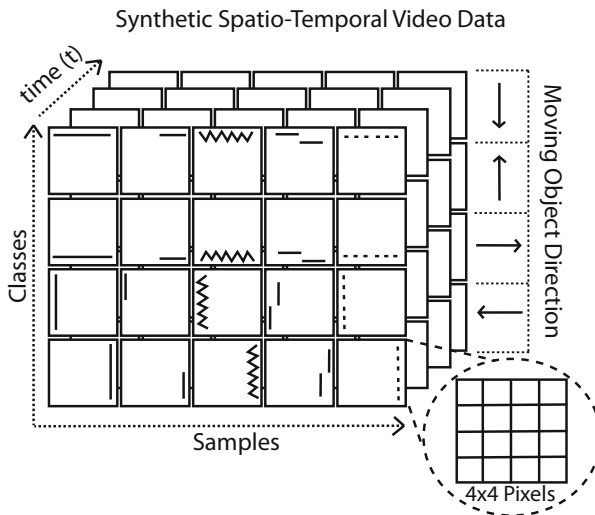

Fig. 2. The figure above illustrated the synthetic video data. There are four classes corresponding to the 4 different directions of their movement where each class consists of 5 samples. The arrow head points towards the direction in which the objects will be moving.

(from up to down, left to right, down to up and right to left). Moreover, from Fig. 2 it can be seen that each of the samples belonging to the same class has varying amount of noise (distorted shapes). There are in total 20 video sequences in the dataset. Each of the videos have a frame rate of 25 frames per second with time span averaging around ≈ 4 seconds. All video sequences are then resized to $4 \times 4 \times 4$.

Since our goal is to apply our method for action recognition of moving objects. This particular synthetic data set was designed to test the systems capability of classifying moving objects based on their trajectory/motion. Furthermore, this synthetic dataset will also confirm the models feasibility in handling continuous spatio-temporal data stream where the epSNNr is provided with multiple spikes as input (i.e. 3 dimensional inputs).

4.3   Design of the Experiment

Similar to [14], we have use the population rank encoding method for transforming the continuous value input variables into spikes. These spikes are then fed to the epSNN reservoir which results in liquid responses.

It can be seen (from Fig. 3) that there are sharp peaks in the peristimulus time histograms (PSTH). This is due to occurrence of spikes after every repetition. These spikes are also know as reliable spikes and are useful for training the algorithms in order to map a particular reservoir response to a desired class label. Figure 3) shows the raster plot and PSTH produced by Step-wise Noisy Threshold probabilistic neuronal model for a particular instance belonging to the four different classes. On acquiring these liquid responses from the last layer of epSNN reservoir, they are concatenated as state vectors according to their corresponding classes. On transforming these liquid responses to state vectors, they are used for training and testing the classifiers.

For our pilot experiment, we have used 5 different types of classifiers as the readout functions which are namely Naivebayes, Multi-Layered Perceptron (MLP), Radial Basis Function (RBF), Decision Tree Induction Algorithm (J48) and Support Vector Machine (SVM). Default parameter settings were used for each of the classifiers in all our experiments. For MLP, the learning rate has been set to 0.3, with 64 hidden nodes for 500 epochs. RBF kernel was used for SVM with gamma value as 0.0 and weights as 1. As for the J48, the confidence factor used for pruning is 0.25 and the minimum number of instances per leaf is set to 2.

Due to the sparsity of the data samples in each class, we have used the leave-one-out cross-validation method for the training and testing of all the five classifiers. This allows us to test all the samples while being unbiased and with minimum variance. The experiment was run 10 times and the obtained test results are averaged. Moreover, no pre-processing steps such as feature selection were applied on the synthetic video dataset.

Since, one of the purposes of this study is to investigate the feasibility of epSNNr for spatio-temporal video pattern recognition using different probabilistic neuron models. We have tested our synthetic video dataset with three
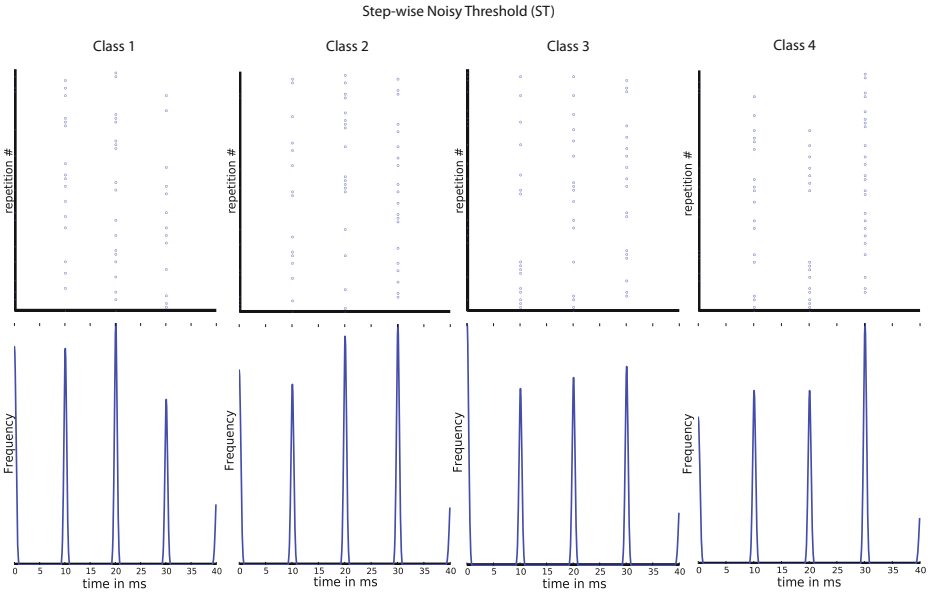
Fig. 3. The figure shows the raster plots and PSTH of 4 typical states for the 4 classes produced by Step-wise Noisy Threshold (ST). The top row shows the raster plot of the neural response of epSNNr with ST probabilistic neurons recorded in 64 repetitions. The bottom row presents the corresponding smoothed PSTH for each raster plot. Each column corresponds to 4 different classes as indicated by the plot labels.

probabilistic neuron models namely, Noisy Reset (NR), Step-wise Noisy Threshold (ST) and Continuous Noisy Threshold (CT) along with the standard Leaky Integrate and Fire (LIF) neuron model. In order to continuously feed three dimensional inputs to the reservoir, the dimensions of the input layer are set as 4×4. This input layer dimensions are the same as that of the synthetic video data. Therefore, there is one input neuron for each pixel at a time.

4.4   Experimental Results

In order to evaluate epSNNr's performance on different classifiers, the state of the epSNN 'reservoir' $S(t)$ is captured in an output module. These captured liquid state $S(t)$ are then used for training and testing the classifiers. Similarly, the performances of all the classifiers were also tested without the epSNNr.

From table 1, it can be seen that epSNNr approach is more accurate and flexible than using standard SNN. Also on an average, the probabilistic neuronal models performed 7.09% better than the traditional deterministic LIF neuron model. Furthermore, when compared to the results obtained by the classifiers without the reservoir, the epSNNr approach average performance was 37.55% higher. We assume that this is due the epSNNr's ability to naturally process spatio-temporal data streams when compared to traditional methods. Also, the

Table 1. The following table presents the Classification Accuracy (Acc.) and Standard Deviation (Std. Dev.) for 5 different methods namely Naivebayes, Multi-Layered Perceptron (MLP), Radial Basis Function (RBF), J48 Decision Tree and Support Vector Machine (SVM)

Methods (Classifiers)	Without Reservoir Acc.(%)/Std. Dev.	With Reservoir			
		LIF Model	NR Model	ST Model	CT Model
Naivebayes	36.45 ± 08.3073	48.92 ± 11.3356	65.00 ± 09.4786	75.00 ± 22.9640	78.39 ± 06.6023
MLP	50.00 ± 15.9344	98.75 ± 02.7951	100.00 ± 0.0000	100.00 ± 0.0000	100.00 ± 0.0000
RBF	55.00 ± 08.1490	93.75 ± 10.8253	96.25 ± 05.5902	96.25 ± 03.4233	93.75 ± 06.2500
J48	36.25 ± 06.8465	53.57 ± 17.0240	63.60 ± 11.9486	61.25 ± 16.7705	63.92 ± 17.2511
SVM	46.25 ± 12.1835	81.25 ± 19.2638	80.10 ± 19.3137	83.75 ± 17.4553	77.50 ± 18.0061

probabilistic neuron models further enhance the separability of the reservoir. The advantage of probabilistic neural model has been well established in previous studies [15] and it is also apparent from our experiment. From table 1, it can be seen that our proposed epSNNr approach performs very well especially with classifiers such as MLP and RBF for this particular dataset.

5   Conclusion and Future Works

This particular pilot study shows epSNNr's capability of handling continuous multiple spike injection using probabilistic neuron model. Moreover, the difference in the recognition rates for the system when compared to the results obtained by the classifiers without the reservoir, the epSNNr approach average performance was significantly higher. This proves that the use of probabilistic neuronal models is superior in several aspects when compared with the traditional deterministic SNN models, including a better performance on noisy data. However, further study on the behavior of the epSNNr architecture under different conditions is needed and more experiments are required to be carried out on benchmark action recognition video datasets.

Several methods will be investigated for the improvement of the epSNNr: Using dynamic selection of the 'chunk' of input data entered into the epSNNr; A new algorithm for an evolving (adaptive) learning in the epSNNr will be developed. In order to improve the separability of the reservoir, we shall experiment with Separation Driven Synaptic Modification (SDSM) approach that has been proposed by [18]. With this approach the viscosity of the reservoir is adjusted by modifying the synapses of the network. Moreover, it has been well established that there is a high correlation between accuracy and separability, hence high separability translates to higher accuracy [18].

Using more complex probabilistic spiking neuron models, such as [13], would require dynamic optimization of its probabilistic parameters. We intend to use a gene regulatory network (GRN) model to represent the dynamics of these parameters in relation to the dynamics of the spiking activity of the epSNNr as suggested in [20]. Each of the probability parameter, the decay parameter, the

threshold and other parameters of the neurons, will be represented as a function of particular genes for na set of genes related to the epSNN model, all genes being linked together in a dynamic GRN model. Furthermore, various parameters such as the connection probability, size and shape of the network topology shall also be tested. In this respect the soft winner-take-all topology will be investigated [21]. For applications that require on line training we intend to use evolving SNN classifier [11],[12]. Finally, implementation of the developed models on existing SNN hardware [22],[23] will be studied especially for on-line learning and object recognition applications such as intelligent mobile robots [24].

References

1. Rabiner, L.: A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE 77(2), 257–286 (1989)
2. Trentin, E., Gori, M.: A survey of hybrid ANN/HMM models for automatic speech recognition. Neurocomputing 37(1-4), 91–126 (2001)
3. Poppe, R.: A survey on vision-based human action recognition. Image and Vision Computing 28(6), 976–990 (2010)
4. Turaga, P., Chellappa, R., Subrahmanian, V., Udrea, O.: Machine recognition of human activities: A survey. IEEE Transactions on Circuits and Systems for Video Technology 18(11), 1473–1488 (2008)
5. Arel, I., Rose, D., Karnowski, T.: Deep Machine Learning: A New Frontier in Artificial Intelligence Research [Research Frontier]. IEEE Computational Intelligence Magazine 5(4), 13–18 (2010)
6. Gerstner, W., Kistler, W.: Spiking neuron models: Single neurons, populations, plasticity. Cambridge University Press (2002)
7. Maass, W., Markram, H.: Synapses as dynamic memory buffers. Neural Networks 15(2), 155–161 (2002)
8. Legenstein, R., Naeger, C., Maass, W.: What can a neuron learn with spike-timing-dependent plasticity?. Neural Computation 17(11), 2337–2382 (2005)
9. Maass, W., Natschläger, T., Markram, H.: Real-time computing without stable states: A new framework for neural computation based on perturbations. Neural Computation 14(11), 2531–2560 (2002)
10. Natschlager, T., Maass, W.: Spiking neurons and the induction of finite state machines. Theoretical Computer Science 287(1), 251–265 (2002)
11. Kasabov, N.: Evolving connectionist systems: the knowledge engineering approach. Springer-Verlag New York Inc. (2007)

12. Schliebs, S., Kasabov, N., Defoin-Platel, M.: On the Probabilistic Optimization of Spiking Neural Networks. International Journal of Neural Systems 20(6), 481–500 (2010)
13. Kasabov, N.: To spike or not to spike: A probabilistic spiking neuron model. Neural Networks 23(1), 16–19 (2010)
14. Wysoski, S., Benuskova, L., Kasabov, N.: Evolving spiking neural networks for audiovisual information processing. Neural Networks 23(7), 819–835 (2010)
15. Schliebs, S., Nuntalid, N., Kasabov, N.: Towards Spatio-Temporal Pattern Recognition Using Evolving Spiking Neural Networks. In: Wong, K.W., Mendis, B.S.U., Bouzerdoum, A. (eds.) ICONIP 2010, Part I. LNCS, vol. 6443, pp. 163–170. Springer, Heidelberg (2010)
16. Hamed, H., Kasabov, N., Shamsuddin, S.: Probabilistic evolving spiking neural network optimization using dynamic quantum-inspired particle swarm optimization. Australian Journal of Intelligent Information Processing Systems 11(1) (2010)
17. Verstraeten, D., Schrauwen, B., D'Haene, M., Stroobandt, D.: An experimental unification of reservoir computing methods. Neural Networks 20(3), 391–403 (2007)
18. Norton, D., Ventura, D.: Improving the separability of a reservoir facilitates learning transfer. In: Proceedings of the 2009 International Joint Conference on Neural Networks, pp. 544–549. IEEE Press (2009)
19. Maass, W., Zador, A.: Computing and learning with dynamic synapses. Pulsed Neural Networks, 157–178 (1999)
20. Kasabov, N., Schliebs, R., Kojima, H.: Probabilistic Computational Neurogenetic Framework: From Modelling Cognitive Systems to Alzheimer's Disease. IEEE Transactions on Autonomous Mental Development (2011)
21. Rutishauser, U., Douglas, R., Slotine, J.: Collective Stability of Networks of Winner-Take-All Circuits. Neural Computation, 1–39 (2011)
22. Indiveri, G., Stefanini, F., Chicca, E.: Spike-based learning with a generalized integrate and fire silicon neuron. In: Proceedings of 2010 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1951–1954. IEEE (2010)
23. Indiveri, G., Chicca, E., Douglas, R.: Artificial cognitive systems: From vlsi networks of spiking neurons to neuromorphic cognition. Cognitive Computation 1(2), 119–127 (2009)
24. Bellas, F., Duro, R., Faina, A., Souto, D.: Multilevel Darwinist Brain (MDB): Artificial Evolution in a Cognitive Architecture for Real Robots. IEEE Transactions on Autonomous Mental Development 2(4), 340–354 (2010)
25. Goodman, D., Brette, R.: The Brian simulator. Frontiers in Neuroscience 3(2), 192 (2009)

Nonlinear Effect on Phase Response Curve of Neuron Model

Munenori Iida[1,2], Toshiaki Omori[1,3], Toru Aonishi[4,3], and Masato Okada[1,3]

[1] Graduate School of Frontier Sciences, The University of Tokyo, Japan
[2] Research Fellow, Japan Society for the Promotion of Science, Japan
[3] Brain Science Institute, RIKEN, Japan
[4] Interdisciplinary Graduate School of Science and Engineering,
Tokyo Institute of Technology, Japan

Abstract. One of the more useful tools for better understanding population dynamics is the phase response curve (PRC). Recent physiological experiments on the PRCs using real neurons showed that different shapes of the PRCs are generated depending on the perturbation, which has a finite amplitude. In order to clarify the origin of the nonlinear response of the PRCs, we analytically derived the PRCs from single neurons by using a spike response model. We clarified the relation between the subthreshold membrane response property and the PRC. Furthermore, we performed numerical simulations using the Hodgkin-Huxley model and their results have shown that a nonlinear change of the PRCs is generated. Our theory and numerical results imply that the nonlinear change of PRCs is due to the nonlinear element in spike time shift of firing neurons induced by the finite amplitude of the perturbation stimuli.

Keywords: phase response curve, spike response model, neuron model.

1 Introduction

One of the more useful tools for better understanding population dynamics is the phase response curve (PRC) [1,2]. We are now able to discuss the responses of oscillatory neurons to external perturbation stimuli without knowing the inner state of those neurons by using a phase reduction method.

Recent physiological experiments [3,4,5] on the PRCs using real neurons have found that different shapes of the PRCs are generated depending on the perturbation, which has a finite amplitude. It is hard to express the nonlinear changes of the PRCs depending on the amplitudes of the perturbation stimuli when using the conventional theory for PRCs [1,2] because the amplitudes of the perturbation stimuli are assumed to be negligibly small. Therefore, the PRCs must be derived while taking into account the nonlinearity against the perturbation stimuli, which has a finite amplitude [3,4,5] .

We would like to figure out what triggers the nonlinearity change of the PRCs against the perturbation stimuli, which have different amplitudes. We consider there are a couple of possible reasons for this change. One is the nonlinearity

caused by the impulse responses of the neurons, which depend on the amplitude of the perturbation stimuli. The other is the nonlinearity caused by the shift in spike time of the oscillatory neurons, which depend on the amplitude of the perturbation stimuli. However, it is unclear how these causes influence the nonlinear change of the PRCs depending on the perturbation stimuli whose amplitude is different. We thus are trying to clarify the cause of the change of the PRCs depending on the amplitude of the perturbation stimuli by using a mathematical neuron model.

In this study we analytically derive the PRCs with a nonlinearity against the perturbation stimuli, which has a finite amplitude, by using the spike response model [6,7], and discuss the influence of the amplitude of the perturbation stimuli on the PRCs. In the SRM, the membrane response property found in the real neurons is described by the kernels. We are able to handle the SRM using biological plausibility by appropriately setting the kernel. One of the most biologically plausible models is the Hodgkin-Huxley model (HHM) [8], but it is not analytically tractable because it is described by using simultaneous nonlinear differential equations. We thus numerically obtained the PRCs from the HHM. We show the influence of the nonlinearity against the perturbation stimuli whose amplitude are finite on the PRCs by comparing the PRCs obtained analytically from the SRM with the PRCs obtained numerically from the HHM. Our results showed that the main cause of the nonlinear change in the PRCs depending on the amplitude of perturbation stimuli is due to the nonlinear element in spike time shift of firing neurons induced by the amplitude of the perturbation stimuli.

This paper is organized as follows. In Sec. 2 we define the SRM and the PRC and we derive the PRC of the SRM. In Sec. 3 we show the results of our numerical simulations using different amplitudes of perturbation stimuli, and in Sec. 4 we make our concluding remarks.

2   Formulation

2.1   Spike Response Model

We describe the dynamics of a spike response model (SRM) [6,7] in this subsection. The membrane potential of a neuron $u(t)$ at time t in the SRM is described by,

$$u(t) = \eta(t - t^f) + \int_0^{t-t^f} \kappa(s) I_{\text{ext}}(t - s) \mathrm{d}s. \tag{1}$$

The neuron fires when the membrane potential $u(t)$ reaches the given threshold. The time t^f denotes the recent fired time in Eq. (1). We set the reference potential of the membrane potential $u(t)$ as the resting potential of the neuron.

The membrane potential $u(t)$ in Eq. (1) consists of two terms in the SRM. The first term of Eq. (1) expresses the effect of the action potential of the neuron. The action potential is a short electrical pulse when the neuron fires. We show the kernel $\eta(t - t^f)$ in Fig. 1 in this paper. We add a perturbation stimulus

at time t_0 to the neuron, which then takes on the resting potential, as shown in Fig. 1. The neuron fires when the amplitude of the perturbation stimulus is sufficiently large. We define the kernel $\eta(t - t^f)$ as the impulse response of the membrane potential when the neuron fires.

The second term in Eq. (1) expresses the influence of the external inputs on the membrane potential. When we add an external current $I_{\text{ext}}(t - s)$ to the neuron at time $t - s$, the membrane potential increases by $\kappa(s)\, I_{\text{ext}}(t - s)\mathrm{d}s$. The kernel $\kappa(s)$ is a proportionality factor for the external currents. The dynamics of the sub-threshold membrane potential are described by using a convolution function of the kernel $\kappa(s)$ and the external inputs $I_{\text{ext}}(t - s)$. This kernel $\kappa(s)$ is presented in Fig. 2. We add a perturbation stimulus with sufficiently small amplitude to the neuron, which takes on the resting potential at time t, as shown in Fig. 2. This perturbation stimulus causes a change in the dynamics of the membrane potential. When the amplitude of the perturbation stimulus is a finite value, the membrane potential of a neuron does not fire and thus decays to the resting potential. We define the subthreshold dynamics of the membrane potential as $\kappa(s)$.
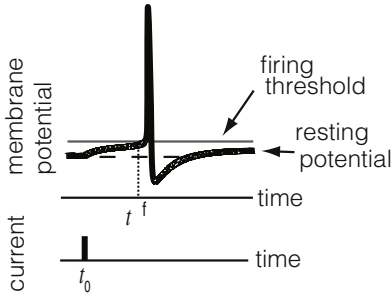

Fig. 1. Schematic of kernel η. A kernel η is obtained from a action potential (top), which is caused by a perturbation stimulus (bottom) whose amplitude is sufficiently large at time t_0.
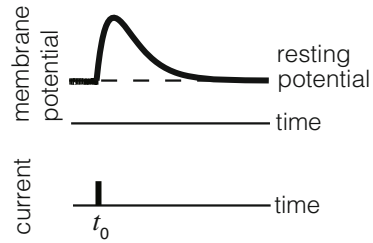
Fig. 2. Schematic of kernel κ. A kernel κ is obtained for an impulse response of membrane potential (top) that is caused by a perturbation stimulus (bottom) whose amplitude is a sufficiently small at time t_0.

We assume the external input $I_{\text{ext}}(t)$ is $I_{\text{ext}}(t) = I_0 + \epsilon\delta(t - t_0)$, by using a constant current I_0 and a perturbation stimulus $\epsilon\delta(t - t_0)$, where $\delta(\cdot)$ is a Dirac's delta function, and the constant input I_0 has a constant value. The perturbation stimulus $\epsilon\delta(t - t_0)$ is added to a neuron at time t_0, where the amplitude of the stimulus is denoted by ϵ.

The dynamics of the membrane potential are characterized by the kernel $\eta(t - t^f)$ and $\kappa(s)$ in the SRM [7]. The kernels biologically correspond to the impulse responses of the membrane potential in real neurons and those in conductance-based neuron models [6,7]. The SRM reproduces the responses of the membrane potential of each neuron by relating the membrane potential dynamics of real neurons to the kernel function of SRM.

2.2   Analytical Derivation of PRC Using SRM

Phase Response Curve. We describe the phase response curve (PRC) [1,2] in this subsection. The PRC characterizes how sensitively oscillatory neurons respond to external perturbation stimuli. We assume in this paper that neurons periodically fire when induced by a constant current. When we use the external $I_{\text{ext}}(t) = I_0$, the constant current I_0 induces the oscillatory firing on neurons. We add the perturbation stimuli to neurons that periodically fire. We then obtain the PRCs of neurons from the spike time shift depending on the stimulated time and the stimulus amplitude.
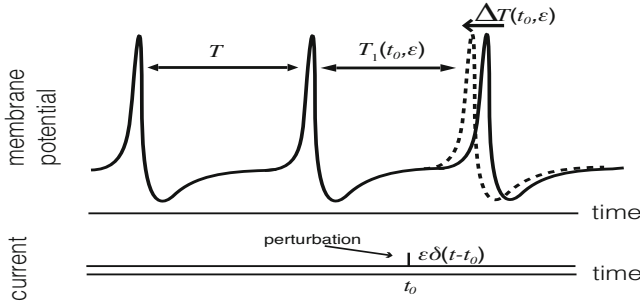

Fig. 3. Schematic of response of model neuron. The regular spikes (top, bold line) of the membrane potential are induced by the external input, which has a constant value. The perturbation response of the neuron, which spikes periodically (top, dashed line), is caused by the constant current and the perturbation stimulus whose amplitude is ϵ. We obtain a phase response curve (PRC) Z from the spike time shift $\Delta T(t_0, \epsilon)$.

We explain the external inputs and the membrane potential dynamics of a model neuron using Fig. 3. The bottom of Fig. 3 shows the external inputs. The top part of Fig. 3 shows the time course of the membrane potential. When the external input is a constant current ($\epsilon = 0$), we set the dynamics of membrane potential to $u_0(t)$,

$$u_0(t) = \eta(t - t^f) + I_0 \int_0^{t-t^f} \kappa(s)\mathrm{d}s. \tag{2}$$

The solid line in Fig. 3 denotes the membrane potential $u_0(t)$ when added with the no perturbation stimulus ($\epsilon = 0$). We set the periodic cycle of the membrane potential $u_0(t)$ at T. The dashed line in Fig. 3 denotes the membrane potential $u(t)$ when added with the perturbation stimulus ($\epsilon \neq 0$).

As shown by the dashed line at the top in Fig. 3, the perturbation stimulus $\epsilon\delta(t - t_0)$ induces the time course $u(t)$, which is different from $u_0(t)$. When we add the perturbation stimulus $\epsilon\delta(t - t_0)$ to the neuron, we set the next fired time at $T_1(t_0, \epsilon)$. The time $T_1(t_0, \epsilon)$ depends on the perturbation time t_0 and its amplitude ϵ. We define the time shift on the neuron induced by the added perturbation stimulus as

$$\Delta T(t_0, \epsilon) = T - T_1(t_0, \epsilon). \tag{3}$$

The amplitude of the perturbation stimulus ϵ has a finite value in biological experiments [3,4,5]. We expand the time shift $\Delta T(t_0, \epsilon)$ with respect to the amplitude of the perturbation stimuli ϵ,

$$\Delta T(t_0, \epsilon) = a_0(t_0) + a_1(t_0)\epsilon + a_2(t_0)\epsilon^2 + a_3(t_0)\epsilon^3 + \ldots. \tag{4}$$

We denote the time shift $\Delta T(t_0, \epsilon)$ in this paper by using the two-dimensional polynomial function of the amplitude of the perturbation stimulus ϵ. Here, we set $a_0(t_0) = 0$ because the time shift $\Delta T(t_0, \epsilon)$ is zero when the amplitude of the perturbation stimulus ϵ is zero. We normalize $\Delta T(t_0, \epsilon)$ by using the period T and the amplitude of the perturbation stimulus ϵ, and set the PRC $Z(t_0)$ to

$$Z(t_0) = \frac{\Delta T(t_0, \epsilon)}{T\epsilon}. \tag{5}$$

We define the PRC $Z(t_0)$ as follows,

$$Z(t_0) = z_l(t_0) + z_n(t_0)\epsilon. \tag{6}$$

Here, we used the following equations in Eq. (6),

$$z_l(t_0) = \frac{a_1(t_0)}{T}, \qquad (7) \qquad\qquad z_n(t_0) = \frac{a_2(t_0)}{T}. \qquad (8)$$

Equations (7) and (8) denote the linear element $z_l(t_0)$ and the nonlinear element $z_n(t_0)$ of the PRC against the applied perturbation stimuli whose amplitude is ϵ, respectively. The linear element $z_l(t_0)$ corresponds to the infinitesimal PRC (iPRC) [1,2]. The existing PRCs theories are mostly discussed using iPRC. The nonlinear element $z_n(t_0)$ is an element of phase response $Z(t_0)$, which cannot be captured with only the linear response. We discuss the nonlinearity of the PRCs against the perturbation stimuli by showing $z_n(t_0)$.

Analytical Derivation of PRC Using SRM. In this subsection, we show how to analytically derive the PRC of the SRM. We focus on the firing threshold and the membrane potential of neurons that periodically fire. Figure 4 shows the schematic diagram used to derive the PRC of the spike response model.

The solid line in Fig. 4 denotes the time course of membrane potential $u_0(t)$ without perturbation stimuli ($\epsilon=0$). The dashed line in Fig. 4 denotes the time course of membrane potential $u(t)$ with perturbation stimuli ($\epsilon \neq 0$). The membrane potential $u(t)$ shifts only $\epsilon\kappa(t-t_0)$ from the membrane potential $u_0(t)$ due to the perturbation stimuli $\epsilon\delta(t - t_0)$. From Eqs. (1) and (2), we obtain

$$u(t) = u_0(t) + \epsilon\kappa(t - t_0). \tag{9}$$

We also discuss the membrane potential at a spiked time. The dot-dashed line in Fig. 4 denotes the firing threshold. As shown in Fig. 3, the membrane potential $u(t)$ reaches the firing threshold at time T_1 and the membrane potential $u(T_1)$ is the firing threshold. Thus, at spiked time T_1, we obtain the following equation,
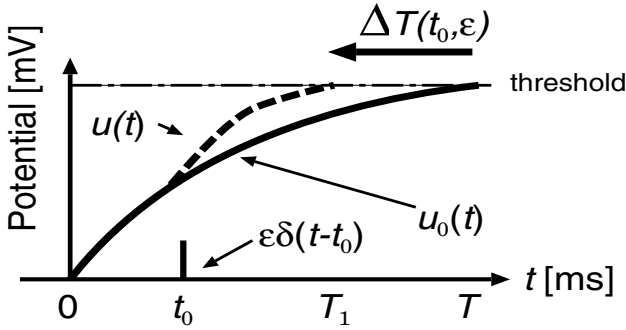
Fig. 4. Schematic diagram used to theoretically derive PRC of spike response model. The time course of a membrane potential without perturbation is shown by $u_0(t)$ (solid line), and the time course of a membrane potential induced by perturbation at time t_0 is shown by $u(t)$ (dashed line). The dashed-dotted line denotes the firing threshold. The period and the spike time shift is denoted by T and $\Delta T(t_0, \epsilon)$, respectively. We introduce more information in the main paragraph.

$$u(T_1) = u_0(T_1) + \epsilon\kappa(T_1 - t_0). \tag{10}$$

The spiked time is denoted $T_1 = T - \Delta T(t_0, \epsilon)$ by using the spike time shift $\Delta T(t_0, \epsilon)$. Therefore, we obtain

$$u(T_1) = u_0(T - \Delta T(t_0, \epsilon)) + \epsilon\kappa(T - \Delta T(t_0, \epsilon) - t_0). \tag{11}$$

On the other hand, the membrane potential $u_0(t)$ without perturbation stimuli reaches the firing threshold every period T. Since the membrane potential $u_0(T)$ at time T is the firing threshold, we obtain $u_0(T) = u(T_1)$. Thus, we obtain as follows,

$$u_0(T) = u_0(T - \Delta T(t_0, \epsilon)) + \epsilon\kappa(T - \Delta T(t_0, \epsilon) - t_0). \tag{12}$$

We expand the first and second terms of Eq. (12) as follows,

$$u_0(T - \Delta T(t_0, \epsilon)) = u_0(T) - \left(\frac{\mathrm{d}}{\mathrm{d}t}u_0(t)\Big|_{t=T}\right)\Delta T(t_0, \epsilon) + \cdots, \tag{13}$$

$$\kappa(T - \Delta T(t_0, \epsilon) - t_0) = \kappa(T - t_0) - \left(\frac{\mathrm{d}}{\mathrm{d}s}\kappa(s)\Big|_{s=T-t_0}\right)\Delta T(t_0, \epsilon) + \cdots. \tag{14}$$

We substitute $\Delta T(t_0, \epsilon) = T\left(z_l(t_0)\epsilon + z_n(t_0)\epsilon^2\right)$, which is obtained from Eqs. (5) and (6), and organize it as ϵ, and then obtain the following,

$$Tu^{(1)}z_l(t_0) - \kappa^{(0)}(t_0) +$$
$$\left(Tu^{(1)}z_n(t_0) - u^{(2)}z_1(t_0)^n + \kappa^{(1)}(t_0)z_l(t_0)\right)\epsilon + \mathrm{O}(\epsilon^2) = 0 \tag{15}$$

where $\mathrm{O}(\epsilon^2)$ denotes the second or more dimensions of ϵ.

Here, we use the following notations:

$$u^{(1)} = \left.\frac{\mathrm{d}}{\mathrm{d}t}u_0(t)\right|_{t=T}, \quad (16) \qquad u^{(2)} = \left.\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}t^2}u_0(t)\right|_{t=T}, \quad (17)$$

$$\kappa^{(0)}(t_0) = \kappa(T - t_0), \quad (18) \qquad \kappa^{(1)}(t_0) = \left.\frac{\mathrm{d}}{\mathrm{d}s}\kappa(s)\right|_{s=T-t_0}. \quad (19)$$

Therefore, we obtain $z_l(t_0)$ and $z_n(t_0)$ as

$$z_l(t_0) = \frac{\kappa^{(0)}(t_0)}{Tu^{(1)}}, \quad (20)$$

$$z_n(t_0) = \frac{\kappa^{(0)}(t_0)}{Tu^{(1)}}\left(\frac{u^{(2)}\kappa^{(0)}(t_0) - u^{(1)}\kappa^{(1)}(t_0)}{(u^{(1)})^2}\epsilon\right). \quad (21)$$

Thus, we obtain the PRC $Z(t_0)$ as,

$$Z(t_0) = \frac{\kappa^{(0)}(t_0)}{Tu^{(1)}}\left(1 + \frac{u^{(2)}\kappa^{(0)}(t_0) - u^{(1)}\kappa^{(1)}(t_0)}{(u^{(1)})^2}\epsilon\right). \quad (22)$$

Therefore, we have analytically derived the PRC using the SRM. In this paper, we regard the kernel $\kappa(s)$ as a linear response of the amplitude of the perturbation stimuli ϵ, and are able to similarly discuss when the kernels are dependent on the amplitude of the perturbation stimuli ϵ. The relation between the PRCs and the membrane properties of neurons has been analytically clarified using the PRCs described by the kernels of the SRM.

Moreover, from Eq. (22), when the amplitude of perturbation stimuli ϵ is negligibly small, the PRC $Z(t_0)$ corresponds to the linear response such as iPRC. When the amplitude of perturbation stimuli ϵ is not negligibly small, the PRC $Z(t_0)$ changes depending on its amplitude ϵ.

3   Verification Using Conductance-Based Model

In this section, we investigate the change of the kernels of the SRM and the change in the spike times against the perturbation stimuli by using numerical simulations, respectively, to find the main cause of the nonlinear change of PRCs. We performed numerical simulations on the conductance-based model, and examine the PRCs dependency on the amplitude of the perturbation stimuli in Sec. 3.1. In Sec. 3.2, we compare the PRCs obtained from our theory that were derived in Sec. 2.2 with those that are obtained directly from the conductance-based model, using perturbation stimuli whose amplitudes are different. In Sec. 3.3, we show the difference presented by the PRCs that are obtained from the theory against the perturbation stimuli and the difference presented by the PRCs that are obtained directly from the conductance-based model against the perturbation stimuli. We also investigate the cause of the nonlinear change of the PRCs.

3.1   Derivation of PRC of SRM

Conductance-based models are biological plausible neuron models. However, they are hard to analytically handle because they are nonlinear models. In this paper, we perform the following numerical simulations using the conductance-based models [8,9]. The membrane potential V obeys the following equation:

$$C_m \frac{dV}{dt} = -I_{Na} - I_K - I_L + I_{ext}, \tag{23}$$

where $C_m = 1\mu$ F/cm^2 and I_{ext} are the injected current (in μA/cm^2).

The sodium, potassium and leaky currents of the Hodgkin-Huxley Model (HHM) [8] are expressed by I_{Na}, I_K and I_L in Eq. (23), respectively.

The external input $I_{ext}(t)$ is assumed to be a sum of the constant current and a perturbation stimulus, $I_{ext}(t) = I_0 + \epsilon\delta(t - t_0)$, as given in Sec. 2.2. We added the perturbation stimuli to the neurons and obtained the PRCs of the neurons from the time shift, as shown in Fig. 3.

We obtain the PRCs with the perturbation stimuli that have different amplitudes (ϵ=0.01 and 0.1), respectively, and Fig. 5A shows the PRCs of the SRM. The pluses ($+$) denote the PRCs of the HHM with the perturbation stimuli whose amplitude ϵ is 0.01. The crosses ($\times$) denote the PRCs of the HHM with the perturbation stimuli whose amplitude ϵ is 0.1. The horizontal axis shows the phase $\theta(\theta\in (0.5,1.0])$ when the neuron is perturbed, and a phase θ corresponds to t_0/T, where t_0 and T expresses the perturbed time and the period, respectively. The vertical axis shows the PRC $Z(\theta)$. We show $0.5 < \theta < 1.0$ in Fig. 5A. We find that the PRCs of the HHM at $0.8 < \theta < 1.0$ changes when the amplitude of the perturbation stimuli ϵ changes, which is shown in Fig. 5A. Thus, we find the maximum value of the PRC when the amplitude of the perturbation stimuli increases. We also find the nonlinear dependence of the PRCs on the amplitude of the perturbation stimuli in the numerical simulation using the HHM.
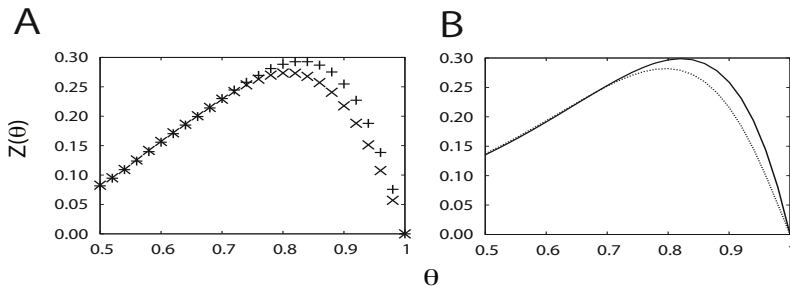

Fig. 5. (A) Numerically obtained PRCs of HHM. The amplitudes of the perturbation stimuli ϵ are $\epsilon = 0.01$ (plus, $+$) and $\epsilon = 0.1$ (cross, $\times$). (B) Theoretically derived PRCs using kernel of SRM that is estimated from HHM. The amplitude of the perturbation stimuli ϵ are $\epsilon = 0.01$ (solid line) and $\epsilon = 0.1$ (dashed line). In A and B, the horizontal axis denotes the phase $\theta(\theta\in (0.5,1.0])$ when the neuron is perturbed, and a phase θ corresponds to t_0/T, where t_0 and T expresses the perturbed time and the period, respectively. The vertical axis shows the PRC $Z(\theta)$.

3.2   Comparison of Derived Theory Using Conductance-Based Models

We compose the PRCs by using the theory presented in Sec. 2.1. Figure 5A and Fig. 5B shows the PRCs that are derived by using the kernels of the SRM. We estimate the kernels of the SRM by using the perturbation stimuli whose amplitude ϵ is 2.1×10^{-3} from the conductance-based model defined in Sec. 3.1.

In Fig. 5B, as with Fig. 5A, we obtain the PRCs when setting the amplitude of the perturbation stimuli (ϵ) at 0.1 and 0.01. The solid line denotes the PRCs of the SRM when the amplitude of the perturbation stimuli ϵ is 0.01. The dashed line denotes the PRCs of the SRM when the amplitude of the perturbation stimuli ϵ is 0.1. We show the PRCs for $0.5 < \theta < 1.0$. We found that the PRCs of the SRM change around $0.8 < \theta < 1.0$ when the amplitude of perturbation stimuli changes. We also confirmed that the maximum value of the PRCs changes when the amplitude of the perturbation stimuli increases. We have verified that the PRCs change depending on the amplitude of the perturbation stimuli using the solid and dashed lines in Fig. 5B.

By paralleling between Figs. 5A and 5B, we confirm that the PRCs obtained directly from the HHM (plus and cross in Fig. 5A) are shaped like the PRCs theoretically derived from the SRM (solid and dash lines in Fig. 5B). The values of the PRCs around phase (θ) $\sim$0.8 in Figs. 5A and 5B are similar. We confirmed through a comparison of Figs. 5A and 5B that the PRCs obtained from an estimation of the kernels are shaped like the PRCs obtained directly from the HHM.

3.3   Dependence of PRC on Amplitude of Perturbation Stimuli

In this subsection, we show the relation between the amplitude of the perturbation stimuli ϵ and the nonlinear dependence of the PRCs on the perturbation stimuli to clarify the relation between the perturbation stimuli and the PRCs. In Fig. 6, the crosses denote the amount of nonlinear dependence of the PRCs of the HHM. The horizontal line in Fig. 6 denotes the amplitude of the perturbation stimuli ϵ. The vertical line is the change amounts in the PRCs or the kernels, which are normalized by the amounts obtained for the perturbation stimuli $\epsilon = 2.1 \times 10^{-3}$. As shown crosses in Fig. 6, the PRCs of the HHM change depending on the amplitude of the perturbation stimuli.

We would like to figure out what triggers the nonlinearity change of the PRCs. As already mentioned in Sec. 1, we consider there are two possible reasons for the nonlinearity change of the PRCs. One is the nonlinearity caused by the kernel of the SRM, which depend on the amplitude of the perturbation stimulus. The other is the nonlinearity caused by the nonlinear elements of the PRCs derived in Eq. (22), which depend on the amplitude of the perturbation stimulus. We investigated the nonlinearity of the kernel of the SRM and the nonlinearity of the PRCs, caused by the perturbation stimuli.

First, we investigated the nonlinearity caused by the kernel of SRM depending on the amplitude of the perturbation stimuli. In Fig. 6, the pluses denote the
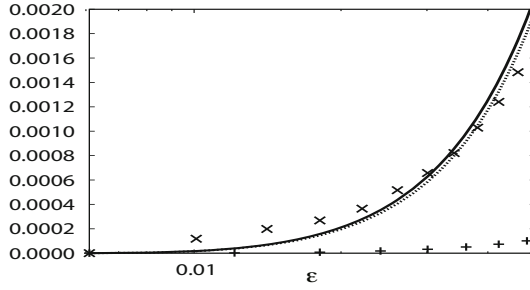
Fig. 6. Change in kernel (plus) and change in PRCs (cross) obtained from numerical simulation using HHM. Perturbation stimuli in each have different amplitude ϵ. The solid and dashed lines denote the change in theoretically derived PRCs using kernel of SRM that is estimated from HHM. We used the kernels estimated by using the perturbation stimuli with fixed (solid line) and changed (dashed line) amplitudes.

change in the kernel κ against the amplitude of the perturbation stimuli. We compare the change in the kernel κ (plus) and the change of the PRCs from HHM (cross) in Fig. 6. This inplies that the nonlinearity change of the kernel κ cannot satisfactorily explain the nonlinearity change of the PRCs of the SRM.

Secondly, we investigated the nonlinear elements of the PRCs $z_n(t_0)$ given in Eq. (21) with and without the influence of the nonlinearity change of the kernel κ, respectively, by using our theory. To begin with, we assumed first case that the nonlinear change of the kernel κ influences the PRCs. We estimated the kernel $\kappa(\epsilon)$ for every perturbation stimuli amplitude. We denote the change in the PRCs with the kernel $\kappa(\epsilon)$ depending on the amplitude of the perturbation stimuli by the dashed line in Fig. 6. Thus, we took into account the nonlinearity of the PRCs against the perturbation stimuli (Eq. (22)) with the nonlinearity of the kernel κ against the perturbation stimuli. We based this on the PRCs with the amplitude of the perturbation stimulus $\epsilon = 2.1 \times 10^{-3}$. We believe that the dashed line in Fig. 6 captures the amounts of the crosses.

On the other hand, we also had to consider that the nonlinear change of the kernel κ does not influence the PRCs. We estimated the kernel $\kappa(\epsilon_0)$ by using just one amplitude of the perturbation stimuli ($\epsilon_0 = 2.1 \times 10^{-3}$). We denote the change in the PRCs with the kernel $\kappa(\epsilon_0)$ depending on the amplitude of the perturbation stimuli by the solid line in Fig. 6. Thus, we had to consider the nonlinearity of the PRCs against the perturbation stimuli (Eq. (22)) without the nonlinearity of the kernel κ against the perturbation stimuli. We believe that the solid line in Fig. 6 captures the amounts of the crosses. We have confirmed that the solid and dashed lines explain the changes of the PRCs, which are directly obtained from the HHM (cross) in Fig. 6.

Therefore, we consider the main cause of the nonlinear change in the PRCs depending on the amplitude of perturbation stimuli, is not the nonlinear response property of $\kappa(\epsilon)$, but is the effect of the nonlinear element of the PRCs $z_n(t_0)$ (Eq. (21)).

4   Concluding Remarks

We analytically derived the PRCs from single neurons by using a spike response model in order to clarify the origin of the nonlinear response of the PRCs. We clarified the relation between the subthreshold membrane response property and the PRC. We performed numerical simulations using our theory and our numerical simulation results have shown that the nonlinear change of the PRCs is possible because of the effect of the nonlinear element of the PRCs $z_n(t_0)$.

References

1. Winfree, A.T.: The Geometry of Biological Time. Springer, N.Y (1980)
2. Kuramoto, Y.: Chemical Oscillations, Waves, and Turbulence. Springer, N.Y (1984)
3. Preyer, A.J., Butera, R.J.: Neuronal Oscillators in Aplysia Californica that Demonstrate Weak Coupling in vitro. Phys. Rev. Lett. 95, 138103 (2005)
4. Tsubo, Y., Takada, T., Reyes, A.D., Fukai, T.: Layer and Frequency Dependencies of Phase Response Properties of Pyramidal Neurons in Rat Motor Cortex. Europ. J. Neurosci. 25, 3429–3441 (2007)
5. Lengyel, M., Kwag, J., Paulsen, O., Dayan, P.: Matching Storage and Recall: Hippocampal Spike Timing-Dependent Plasticity and Phase Response Curves. Nat. Neurosci. 8, 1667–1683 (2005)
6. Kistler, W., Gerstner, W., Hemmen, J.: Reduction of the Hodgkin-Huxley Equations to a Single-Variable Threshold Model. Neural Comput. 9, 1015–1045 (1997)
7. Gerstner, W., Kistler, W.: Spiking Neuron Models. Camb. Univ. Press, UK (2002)
8. Hodgkin, A., Huxley, A.: A Quantitative Description of Membrane Currents and its Application to Conduction and Excitation in Nerve. J.Physi. 117, 500–544 (1952)
9. Wang, X.-J., Buzsaki, G.: Gamma Oscillation by Synaptic Inhibition in a Hippocampal Interneuronal Network Model. J. Neurosci. 16, 6402–6413 (1996)

Modulations of Electric Organ Discharge and Representation of the Modulations on Electroreceptors

Kazuhisa Fujita[1,2]

[1] Department of Computer and Information Engineering,
Tsuyama National Collage of Technology,
654-1 Numa , Tsuyama, Okayama, Japan 708-8506
[2] University of Electro-Communications,
1-5-1 Chofugaoka, Chofu, Tokyo, Japan 182-8585
`kazu@spikingneuron.net`

Abstract. Weakly electric fish can recognize object's parameters, such as material, size, distance and shape, in complete darkness. The ability to recognize these object's parameters is provided by electrosensory system of the fish. The fish generates electric field using its electric organ (EOD: electric organ discharge). An object around the fish distorts the self-generated EOD and make the EOD modulation on fish's body surface. The EOD modulation is converted into firings of electroreceptor afferents on fish's body surface. The fish can extract object's parameters from the firings. In the present study, we investigated features of the EOD modulations including information of object's shape. Therefore we calculated EOD modulations generated by objects that were various shapes and firing patterns of electroreceptors evoked by electric images using computer simulation. We found that the shape of an object near the fish was represented by the maximum of firing rate of the receptor network. However the difference of the maximum of the firing rate between various objects was small when the distance of the object from the fish was more than about 3-4 cm. This result suggested that detection limit of the fish for object's shape would be about 3-4 cm and the limit would be smaller than that of other sensory systems.

Keywords: electrosensory system, electric image, weakly electric fish.

1 Introduction

A weakly electric fish generates electric field around its body by emitting electric organ discharges (EOD). It can detect objects by sensing disturbances in the electric field [3,12]. The ability is called electrolocation. The EOD is either continuous and quasi-sinusoidal ('wave-type species'), or discrete and pulse-like ('pulse-type species'). We investigated electrolocation in a wave-type species, *Eigenmannia*. The fish can recognize various object's parameters of an object,

size, distance, shape, electric properties, etc, in complete darkness on electroloca-
tion [18]. Objects with electric properties that differ from those of the surround-
ing water distort the EOD around the fish's body [16]. The level of distortion
depends on the size, lateral distance, shape, and electric properties of the object.
The fish is able to detect the current level using an array of electroreceptors lo-
cated in the skin. The fish must then analyze the two-dimensionally distributed
EOD modulation generated on the skin to locate the object. Thus, an under-
standing of the mechanisms that mediate electrolocation depends on the ability
to quantify the EOD modulation on the body surface.

A number of modeling and theoretical studies have evaluated the EOD mod-
ulation induced by resistive objects [11,13]. The population coding of electrosen-
sory information in electroreceptors and the electrosensory lateral-line lobe was
proposed on the basis of EOD calculations [17,15,7,10,9]. These studies focused
on the changes in amplitude modulation of the EOD caused by an object without
considering various object's properties and evaluated the neuronal mechanisms
underlying amplitude coding in the electrosensory system. Although EOD mod-
ulation induced by shape of an object and mechanism of recognition of various
object's properties is poorly understood. The ability to detect object's proper-
ties is critical as prey such as insect larvae and small crustaceans. However, it is
difficult to accurately measure the spatial distribution of EOD modulation in an
experimental setting. Given this, a theoretical approach may be used to provide
insights into the process.

The EOD modulation is processed by the electroreceptor network on the skin,
the electrosensory lobe in the hind brain and the torus semicircularis in the
midbrain of the fish on electro perception [2]. It is important to clarify the
representation of the EOD modulation induced by an object on response of the
electroreceptor network because extraction of object's parameters on higher elec-
trosensory nucleus is affected by ability of representation of the electroreceptor
network.

In the present study, we focused on one question. What features of the EOD
modulation and response of the electroreceptor network represent object's shape?
To address this issue, we developed a model of the fish body that can be used
to accurately calculate the amplitude and phase modulation of an electric field
produced by objects that had various sizes and shapes. The model is based on
the finite-element-method [13,8]. Furthermore, we made a model of the electrore-
ceptor network. This is the first study to investigate the changes in the EOD
modulation induced by objects with various properties in wave-type fish.

2   Methods

2.1   Simulation Setup

The wave type fish (*Eigenmannia*, etc) was used as a model animal. The fish
size was $16 \times 0.8 \times 3$ cm^3. The fish was kept in the water tank ($24 \times 24 \times 18$ cm^3)
that was filled with water (3.8 kΩcm). The tank was divided into small cube
($0.1 \times 0.1 \times 0.1$ cm^3). An object was near the fish. In the present study, we

used three kinds of objects, the large cubes ($3 \times 3 \times 3$ cm^3), the small cube ($2 \times 2 \times 2$ cm^3), and the pyramid ($3 \times 3 \times 3 \times (1/3)$ cm^3) (shown in Fig. 1B). These objects were metal (0.1 kΩcm). We calculated the EOD with its modulation by an object in order to investigate representation of shape of an object on response of the electroreceptor network of weakly electric fish.
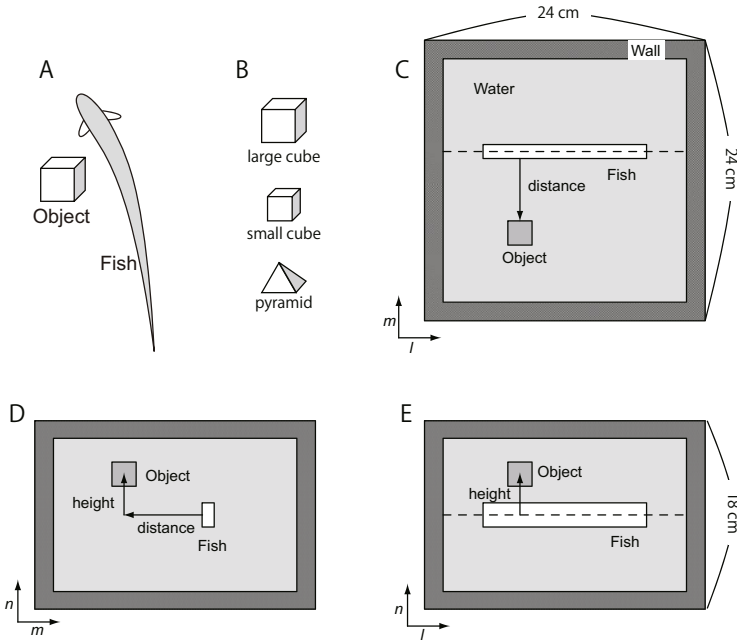

Fig. 1. (A) Situation of the simulation. An object was near the fish. (B) Objects using in the present study. The upper object named "large cube" was the cube ($3 \times 3 \times 3$ cm^3). The middle object named "small cube" was the cube ($2 \times 2 \times 2$ cm^3). The lower object named "pyramid" was the pyramid ($3 \times 3 \times 3 \times (1/3)$ cm^3). (C, D, E) The model for calculating the EOD field of a fish. The fish was in the water tank. In the present study, height of the object was 0 cm (the object is just beside the fish).

2.2   Model for Calculating EOD Modulation Induced by an Object

To calculate EOD field around the fish, we modeled an aquarium (containing the fish), the surrounding medium (water), and an object, in three-dimensional space (Fig. 1C, D, E). We made a model of fish body by which we calculated numerically the spatio-temporal patterns of EOD around the fish. The fish was approximated by a rectangular parallelepiped. The fish was placed in a water container. The model fish consisted of a body surrounded by skin that had high resistivity and an electric dipole produced by the electric organ. The electric organ was represented by a dipole made up of two point sources because the

spatial pattern of the EOD is approximately dipolar at some distance from the point of discharge. The rostral and caudal ends of this organ were placed at (13.6, 14, 9) and (18.4, 14, 9), respectively. Finite-element-method was used to calculate the electrostatic potential at each point on the body and the external space in the case where a metal object was put on a point in the space [13,8]. The detail of the method of the simulation is given in the previous paper [8].

We calculated the voltage across the lateral skin of the fish. The skin is two-dimensional. The transdermal voltage, $V_{\mathrm{skin}i,j}(t)$, at the site of (i,j) on the fish skin is defined by

$$V_{\mathrm{skin}i,j}(t) = A(V_{\mathrm{EOD}} + V_{\mathrm{EODAM}i,j})\sin(2\pi ft + \phi_{\mathrm{EODPM}}), \tag{1}$$

where V_{EOD} is the transdermal voltage without an object, $V_{\mathrm{EODAM}i,j}$ is the EOD amplitude modulation of site (i,j), ϕ_{EODPM} is EOD phase shift, A is the constant value of transdermal voltage, f is the frequency of self-generated EOD.

2.3   Model of the Electrosensory Receptor Network

The model of electrosensory system consisted of two-dimensional electroreceptor network, as shown in Fig. 2. The electroreceptors received EOD and its modulation depending on the position of the electroreceptor on the fish body. We, here, used P-receptors that code the modulation of EOD.
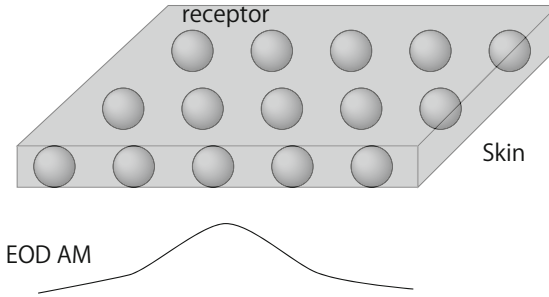

Fig. 2. Our neural model of the electroreceptor network. The receptors were on the fish's skin. The receptors were placed on two dimensional array. The receptors received self-generated EOD and EOD modulation by an object near the fish.

The model of P-electroreceptor and P-afferent nerve was made based on the leaky integrate-and-fire model except that the threshold was also a dynamic variable [5,7]. The dynamic change of the threshold was needed to adaptively regulate the firing rate of P-afferent for various magnitudes of EOD AM stimuli elicited by an object with various distances and shapes. The membrane potential and threshold of P-receptor at (i,j) site, $V_{i,j}$ and $V_{i,j}^{th}$, were determined by

$$\frac{\mathrm{d}V_{i,j}}{\mathrm{d}t} = \frac{1}{\tau_V}(-V_{i,j} + V_{\mathrm{skin}i,j}(t)), \tag{2}$$

$$\frac{dV_{i,j}^{th}}{dt} = \frac{1}{\tau_{th}}(-V_{i,j}^{th} + V_0),$$ (3)

where τ_V and τ_{th} are time constant of $V_{i,j}$ and $V_{i,j}^{th}$, respectively, and V_0 is constant parameter. An action potential is occurred when the membrane potential is equal to the threshold for firing. The threshold is incremented by a constant during the firing of a receptor. $V_{\text{skin},i,j}$ is the amplitude to the transdermal voltage to the receptor at (i,j) site, which is calculated by the transepidermal voltage across the fish skin that is obtained using the EOD calculation described in Sec. 2.2. f is the frequency of the fish's own EOD, $f = 400$ Hz.

3   Results

3.1   EOD Modulations

We calculated EOD modulations induced by three types of objects. The calculated EOD modulation provides accurate inputs to the receptor network in
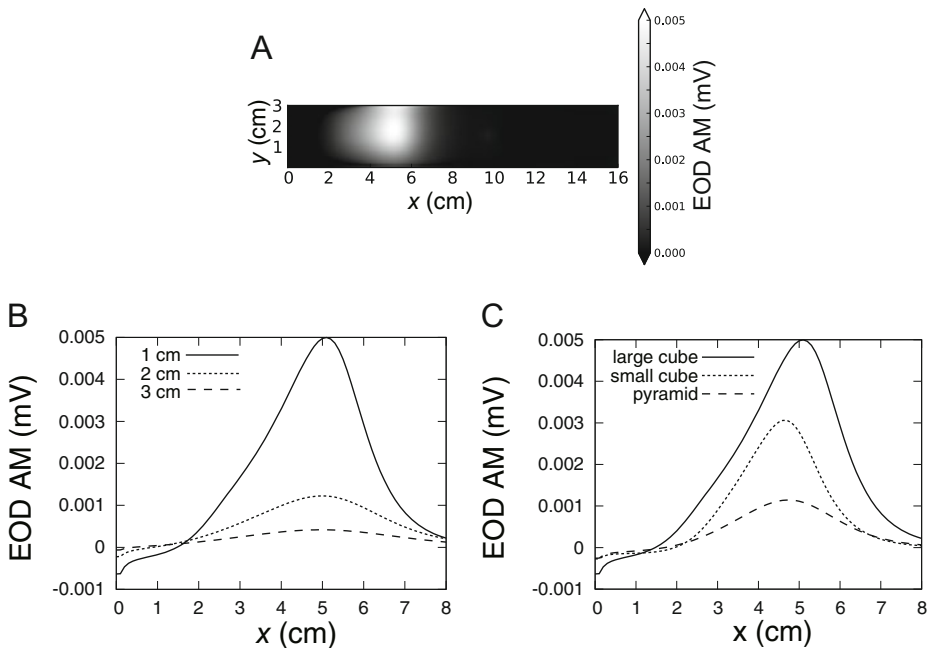

Fig. 3. Spatial distribution of the amplitude modulation of the EOD. (**A**) The EOD amplitude modulations on the skin was generated by the large cube. The EOD modulation two-dimensionally distributed. (**B**) Change in the EOD amplitude modulations (AMs) on the skin generated by the large cube varying distance of the large cube from the fish. (**C**) Change in EOD amplitude modulations on the skin generated by large cube, the small cube and the pyramid when the objects placed at 1 cm from the fish.

order to study the mechanism of information processing of electrorecptors for recognition of the shape of an object.

Figure 3A shows two-dimensionally distributed EOD modulation on the skin of the fish generated by the large cube placed as 1 cm from the fish. Figure 3B shows the change in EOD amplitude modulations induced by the large cube varying the distance of the object from the fish. Figure 3C shows the EOD amplitude modulations induced by the large cube, the small cube and the pyramid placed at 1 cm from the fish. The EOD modulations became smaller and wider as the large cube was moved from the fish (Fig 3B). Conversely, the EOD modulation increased and became wider as the size of the object increased (Fig 3C). The EOD modulation induced by pyramid was smaller and wider than that induced by cubes. These results suggest that the maximum and width of EOD modulation would depend on distance and shape of an object however we could not discriminate between changes in the EOD modulations induced by distance and shape of an object.

3.2   Representation on the Receptor Network

We simulated the response of the receptor network. The receptors were on the skin of the electric fish. The receptor responded to the EOD modulation. The fish has two type receptors. One is the P-type receptor afferent that mainly responds to amplitude modulation. Two is the T-type receptor afferent that mainly responds to the phase of the EOD wave. In the present study, we simulated the response of the P-type electroreceptor network.
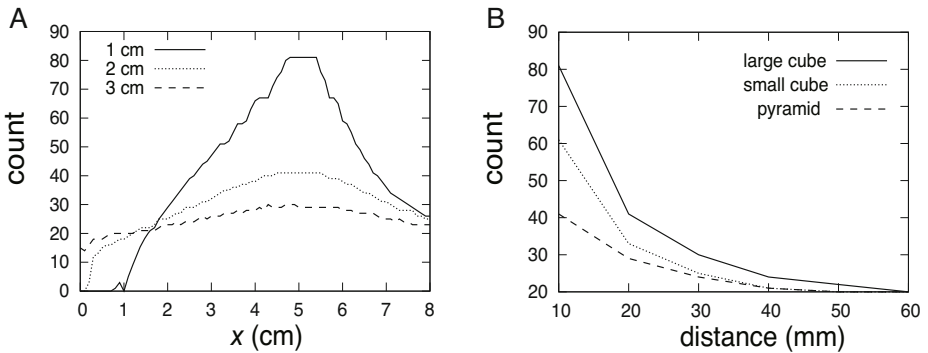
Fig. 4. The response of the receptor network. **A** The responses of the receptor network varying distance of the large cube from the fish. Solid, dotted and broken lines indicate the response of the receptor network stimulated with EOD modulations induced by the large cubes placed at 1, 2 and 3 cm (laterally) from the fish, respectively. **B** The maximum firing rate of the response of the electroreceptor network. Horizontal line indicates distance of the object from the fish. The vertical line indicates the maximum of the firing count of the network for 1000 ms. Solid, dotted and broken lines indicate the maximum firing rate of the receptor network evoked by the large cubes, small cube and pyramid, respectively.

Figure 4A shows firing rates of the receptors located on center line of the network with varying the distance of the large cube from the fish. Spatial distribution of the firing rate resembled "Mexican hat". The mexican hat shape is derived from EOD modulation. Thus the receptor network encoded the shape of the EOD modulation. This result showed the EOD amplitude modulation was represented on the response of the receptor network.

Figure 4B shows the change of maximum firing rate of the response of the receptor network. The maximum of the firing rate depended on the shape of the objects. This result suggests that the maximum of the firing rate would be one of the characteristic features of EOD modulation that represent the shape of an object. However, the difference of the maximum firing rate between objects' shapes was small when distance of the objects from the fish was more than about 3-4 cm. These results show that it may be difficult for the fish to recognize the shape of the object when distance is more than about 3-4 cm because difference of the response of the electroreceptor network did not clearly appear.

4   Concluding Remarks

The purpose of the present study is investigation of difference of the response of the receptor network produced by object's shapes. To address of this issue, we calculated the three-dimensional electric field generated by the weakly electric fish and simulated the response of the receptor network. We found that the shape of an object near the fish was represented by the maximum of firing rate of the receptor network. However the difference of the maximum of the firing rate between various objects was small when the distance of the object from the fish was more than 3-4 cm. In 2010, Von der Emde et al showed the limit of distance of an object on object recognition. The limit is about 4 cm. Our result was in agreement with the result of von der Emde et al [18]. This result suggests that detection limit of the fish for object's shape would be about 3-4 cm and the limit is smaller than that of other sensory systems. In the future work, we will investigate changes in width of the response of the receptor network and index for discrimination between EOD modulation induced by distance, size, shape, etc. Furthermore, we will study the role of spike timing dependent plasticity (STDP) for electrosensory perception because STDP was found in electrosensory lobe, and provides the function of spatial filter [6].

References

1. Bass, A.H.: Electric organs revisited: evolution of a vertebrate communication and orientation organ. In: Bullock, T.H., Heiligenberg, W. (eds.) Electroreception, pp. 13–70. Wiley, New York (1986)
2. Bastian, J.: Electrolocation i. how the electroreceptors of apteronotus albifrons code for moving objects and other electrical stimuli. Annals of the New York Academy of Sciences 188, 242–269 (1971)
3. Bastian, J.: Electroloaction: behavior, anatomy, and physiology. In: Bullock, T.H., Heiligenberg, W. (eds.) Electroreception, pp. 577–612. Wiley, New York (1986)

4. Bennett, M.V.L.: Electric organ. In: Hoar, W.S., Randall, D.J. (eds.) Fish Physiology, vol. 5, pp. 347–491. Academic Press, New York (1971)
5. Brandman, R., Nelson, M.E.: A simple model of long-term spike train regularization. Neural Computaion 14, 1575–1597 (2002)
6. Fujita, K.: Spatial Feature Extraction by Spike Timing Dependent Synaptic Modification. In: Wong, K.W., Mendis, B.S.U., Bouzerdoum, A. (eds.) ICONIP 2010, Part I. LNCS, vol. 6443, pp. 148–154. Springer, Heidelberg (2010)
7. Fujita, K., Kashimori, Y.: Population coding of electrosensory stimulus in receptor network. Neurocomputing 69(10-12), 1206–1210 (2006)
8. Fujita, K., Kashimori, Y.: Modeling the electric image produced by objects with complex impedance in weakly electric fish. Biol. Cybern. 103, 105–118 (2010)
9. Fujita, K., Kashimori, Y., Kambara, T.: Spatiotemporal burst coding for extracting features of spatiotemporally varying stimuli. Biol. Cybern. 97(4), 293–305 (2007)
10. Fujita, K., Kashimori, Y., Zheng, M., Kambara, T.: A role of synchronicity of neural activity based on dynamic plasticity of synapses in encoding spatiotemporal features of electrosensory stimuli. Math. Biosci. 201(1-2), 113–124 (2006)
11. Heiligenberg, W.: Theoretical and experimental approaches to spatial aspects of electrolocation. J. Comp. Physiol. 103, 247–272 (1975)
12. Heiligenberg, W.: Neural nets in electric fish. MIT Press, Cambridge (1991)
13. Hoshimiya, N., Shogen, K., Matsuo, T., Chichibu, S.: The *apteronotus* eod field: Waveform and eod field simulation. J. Comp. Physiol. A 135(4), 283–290 (1980)
14. Knudsen, E.I.: Spatial aspects of the electric fields generated by weakly electric fish. J. Comp. Physiol. 99, 103–118 (1975)
15. Lewis, J.E., Maler, L.: Neuronal population codes and the perception of object distance in weakly electric fish. J. Neurosci. 21(8), 2842–2850 (2001)
16. Lissman, H.W., Machin, K.E.: The mechanism of object location in *gymnarchus niloticus*. J. Exp. Biol. 35, 451–486 (1958)
17. Nelson, M.E., Maclver, M.A.: Prey capture in the weakly electric fish *apteronotus albifrons*: sensory acquisition strategies and electrosensory consequences. J. Exp. Biol. 202(Pt 10), 1195–1203 (1999)
18. von der Emde, G., Behr, K., Bouton, B., Engelmann, J., Fetz, S., Folde, C.: 3-dimensional scene perception during active electrolocation in a weakly electric pulse fish. Frontiers in Behavioral Neuroscience 4 (2010)

Spiking Neural PID Controllers

Andrew Webb, Sergio Davies, and David Lester

School of Computer Science,
The University of Manchester,
Oxford Road, Manchester, UK
{webb,daviess,dlester}@cs.man.ac.uk
http://apt.cs.man.ac.uk/projects/SpiNNaker/

Abstract. A PID controller is a simple and general-purpose way of providing responsive control of dynamic systems with reduced overshoot and oscillation. Spiking neural networks offer some advantages for dynamic systems control, including an ability to adapt, but it is not obvious how to alter such a control network's parameters to shape its response curve. In this paper we present a spiking neural PID controller: a small network of neurons that mimics a PID controller by using the membrane recovery variable in Izhikevich's simple model of spiking neurons to approximate derivative and integral functions.

Keywords: SpiNNaker, neural networks, PID controllers.

1 Introduction

A proportional-integral-derivative (PID) controller is a closed-loop controller that takes the difference between the desired and measured value of some variable of a dynamic system as its input. This difference is the error signal. The controller manipulates one or more inputs to the dynamic system with a control signal to reduce the error. The control signal is produced using a weighted sum of the present, past and predicted future values of the error. Where t is time, $u(t)$ is the control signal and $e(t)$ is the error, a PID controller has the form

$$
\begin{aligned}
P_t &= k_P e(t) \\
I_t &= k_I \int e(t) dt \\
D_t &= k_D \frac{d}{dt} e(t) \\
u(t) &= P_t + I_t + D_t
\end{aligned}
\tag{1}
$$

The advantage of this modular approach is that the parameters k_P, k_I and k_D can be tuned to variously modify the steady-state error, rise time, overshoot, response to step change and settling time of the system. While there are algorithms

to monitor the responsiveness of the system and automatically tune the parameters to optimise control, the controller is simple enough that manual parameter tuning is possible. Conversely, it is difficult to modify the various parameters of a spiking neural network (including individual neuron parameters, synaptic weights and connectivity) and know how this will affect the overall behaviour of the neural network. If we want to make a spiking neural network controller less responsive to noise in the error signal, for instance, it is not clear how to do this.

In this paper we propose a method of building a PID controller using three neurons. The reason for this is to combine the advantages of PID controllers, discussed above, with the advantages of using biologically plausible neural networks for dynamic systems control.

Work has been done to build PID controllers using a collection of artificial neurons [11], using the well known backpropagation algorithm to tune the parameters k_P, k_I and k_D. In this paper we only show that a spiking neural PID controller is possible. Future work will focus on using established, biologically plausible learning algorithms to automatically tune the parameters.

We can build a PID controller using only three spiking neurons, where each neuron approximates one of the terms of the controller, by taking advantage of the spike frequency adaptation behaviour exhibited by biological neurons. Class 1 excitable neurons, when given sufficient current, will spike regularly at a low frequency. This frequency will respond to changes in current according to a continuous F-I curve [7]. Such neurons tend to exhibit spike frequency adaptation behaviour, wherein the neuron habituates to its input. If we introduce a step increase in the input current of a neuron that is firing at a steady rate, the firing rate of the neuron will increase. The neuron will then begin to habituate to the new stimulus; the firing rate will decrease until it settles at a new equilibrium. Similarly, a step decrease in input current will lead to a largely reduced firing rate, which then steadily increases as the neuron habituates. This behaviour is illustrated in fig. 1.
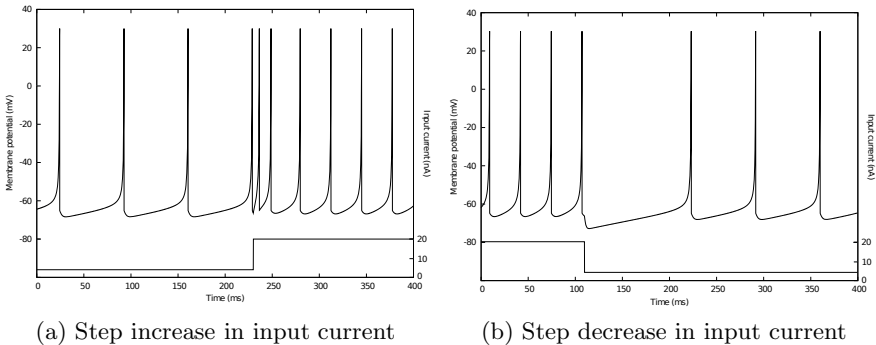

(a) Step increase in input current   (b) Step decrease in input current

Fig. 1. Spike frequency adaptation. The lower line represents the input current of the neuron.

Because of this spike frequency adaptation, the outgoing firing rate of regular spiking class 1 excitable neurons is a function of the input stimulus and the recent rate of change of that stimulus. That is, the firing rate of a neuron is a combination of a proportional and a differential term of the input current, while the relative weights of the two terms depends on the level of adaptation the neuron exhibits. If, for some dynamic system control problem, we provide such a neuron with the error signal - the difference between the measured and desired value of a process variable - as an input current, and use the output to adjust some parameter of the system, then we could consider the neuron to be a simple PD controller. Neurons with stronger habituation would have a higher value for the derivative gain (k_D) and a lower value for the proportional gain (k_P).

By tuning their parameters appropriately, simulated neurons can be made to give outputs that are approximations of each of the functions in (1) applied to the input. The spiking neural PID controller circuit is a collection of such neurons, with the various gain parameters given by the relevant synaptic connections.

2   The SpiNNaker Architecture

Neural simulations were run on the SpiNNaker architecture. SpiNNaker is composed of custom components and off-the-shelf hardware optimized to simulate spiking neural networks [9]. However this architecture does not limit to this task since its main component is an ARM processor with DSP extensions and low power specifications.

The features of this architecture [10] that are relevant to simulate biological structures are:

1. **Native parallelism:** in biological neural networks, each neuron is an individual computational element that receives input from other neurons, creating a massively parallel computational network. The SpiNNaker system mimics this structure using multiple general purpose computational element (18 cores) inside each chip, each with its own private subsystem and shared resources. In particular a communication controller is shared across the chip, which handles the interconnection with other surrounding chips.

2. **Event-driven behaviour:** in biology, neurons communicate through spikes. The SpiNNaker architecture uses AER packets to transmit neural signals between processors (and therefore neurons) over a configurable packet switched network with asynchronous interconnection.

3. **Distributed incoherent memory:** in biology, a central memory does not exist. Neurons use only local information to process incoming stimuli. In the SpiNNaker architecture we use a hierarchy of memories: a Tightly-coupled memory local to to each of the cores and an SDRAM chip shared between the cores in a chip, with an incoherent access mechanism.

4. **Reconfigurability:** in biology, neurons are plastic. This means that the interconnections change both in shape and in strength, while the neural network evolves. Likewise in the SpiNNaker architecture the routing system allows for reconfiguration on-the-fly.

3   Design

We propose a method of combining neurons with different levels of spike frequency adaptation in a small circuit in order to mimic the behaviour of a PID controller. The key to this approach is to treat a spiking neuron as a function which, when applied to an input sequence, gives an output sequence. The goal is to find neural parameters to approximate each of the functions in (1).

The steps in this process are as follows. First, we must choose an appropriate neuron model. Then we must decide on a neural coding method in order to be able to describe neurons as functions of input intensities. Then we use a parameter fitting algorithm to find two sets of neuron parameters. The first neuron must give an output that is proportional to the derivative of the input current. The second neuron must give an output that is proportional to the integral of the input current over some time window. These neurons are then combined in a circuit with a dummy proportional-term neuron to form a spiking neural PID controller, where the weight of each term is determined by a synaptic weight.
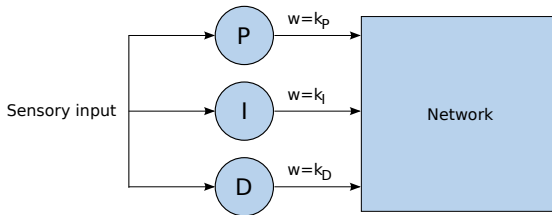

Fig. 2. A spiking neural PID controller

3.1   Neuron Models

In order to simulate spiking neurons we must first choose a neuron model. Some models aim to be biophysically meaningful and are generally computationally expensive. For example, Hodgkin-Huxley type models consist of a set of differential equations and parameters which model the flow of currents through the cell membrane, as well as the opening and closing of ion channels [4]. Other models, which aim for computational efficiency, are less biophysically meaningful. Choosing a neuron model is often a trade-off between these two measures. Any neuron model we use must satisfy two requirements. The model must be relatively computationally simple and it must exhibit spike frequency adaptation behaviour. For our purposes we selected the Izhikevich neuron model. It has the required features and an implementation of the model is currently available on our simulation hardware, the universal real-time spiking neural network architecture, SpiNNaker [2].

Izhikevich's simple model of spiking neurons exhibits a wide range of observed cortical neuron behaviour, including spike frequency adaptation, with a pair of differential equations [5]. The model is extremely computationally simple when compared to other models with as many features [6] and the SpiNNaker implementation of Izhikevich neurons is particularly efficient [8].

The behaviour of Izhikevich's simple spiking neurons is governed by a pair of ordinary differential equations.

$$v' = 0.04v^2 + 5v + 140 - u + I \tag{2}$$

$$u' = a(bv - u) \tag{3}$$

$$\text{if } v \geq 30\text{mV, then } \begin{cases} v \leftarrow c \\ u \leftarrow u + d \end{cases} \tag{4}$$

The variable v represents the membrane potential and the variable u represents the recovery variable, which is responsible for spike frequency adaptation. Of the parameters, a represents the time scale of the recovery variable, b describes the sensitivity of the recovery variable to the value the membrane potential, c is the after-spike reset of the membrane potential, and d is the after-spike reset of the recovery variable. Our parameter fitting algorithm will seek values for these four parameters for three neurons, each of which approximates a function from (1).

3.2   Neural Coding

We need to choose a neural coding method in order to be able to, at a given time, determine the value or intensity of the input or output of a neuron based on the recent spike sequence. We encode the error signal as a current to be injected into each neuron in the circuit, scaled such that it lies in a biologically plausible range.

To decode the output value of a neuron, we take the traditional view that most of the information in a spike train is carried in its mean firing rate. By the most common definition, the mean firing rate of a neuron is taken to be the average spike count over some window [3]. This average is calculated using a causal sliding window, where only past spikes within the window are counted [1]. The averaging window used was 100 ms in length, and was shifted 20 ms at a time, giving (simulation time -100) / 20 data points.

3.3   Approximating the Functions

Proportional Term. The proportional term is given by a regular spiking neuron which exhibits no adaptation behaviour; the parameter d is set to 0.

The input current is scaled such that it lies in a range where the current is always high enough to elicit spikes and always low enough that the spike train is not saturated. That is, the input current is kept in a range where there is a linear relationship between input current and spike frequency.

Derivative Term. If the d parameter is positive (as is usual in models of biological neurons), the neuron has spike frequency adaptation behaviour, as shown in fig. 1. The neuron is essentially subtracting previously seen stimuli from the current stimulus. With a sufficiently high level of adaptation, and a fast-leaking recovery variable, the firing rate of the neuron will be proportional to the current stimulus minus any stimuli seen very recently; the neuron will approximate differentiation of the input. Altering the a parameter will change the window size over which differentiation is approximated. Note that an increase in window size must be offset by a decrease in the parameter d (to give a lower weight to stimuli within the window).

Integral Term. If the d parameter is negative, the neuron has negative spike frequency adaptation behaviour. There is a positive feedback loop, such that every time the neuron spikes the input current required to elicit a spike decreases. If such a neuron is given a constant current, its firing rate will increase over time. The neuron essentially approximates the integral of its input over time. To bound the window over which the neuron integrates, and to stop runaway epileptic behaviour, the recovery variable must leak fairly quickly; again, altering the a parameter changes the size of the window for integration. Altering the d parameter changes the weight given to stimuli within the window. Appropriate values for both parameters will give a neuron whose firing rate is a leaking integral of its input.

3.4   Parameter Fitting Algorithm

To find neurons that approximate differentiation and integration, a parameter fitting algorithm is used, which works as follows. First we construct an input current sequence i, where each value represents an input current for a 5 ms duration. The current sequence, which is to be injected into every trial neuron, is a sinusoid of varying amplitude and frequency. A discrete form of the relevant function from (1) is applied to i to produce i'. In the case of the integral, a leaking integral is calculated.

The search takes place over the Izhikevich neuron parameters a, b, c and d. The input current sequence is injected into a trial neuron for each set of parameters. For each trial neuron, the averaging method outlined in 3.2 is used to construct a sequence of firing rates, f, from the resultant spike train. We interpolate values in the sequence f, such that there is a corresponding value for each value in i', and then calculate the Pearson product-moment correlation coefficient (PMCC) between f and i'. The PMCC is a measure of linear

dependence between two variables, giving values between -1 and 1 inclusive. It is given by the covariance of the two variables divided by the product of their standard deviations. A correlation coefficient of 1 means that a linear equation precisely describes the relationship between the two variables. A coefficient of 0 indicates that there is no linear relationship between the two variables. The neuron with the highest correlation coefficient between its transformed input current and its spike frequency best approximates the relevant function.

4 Results

Table 1 shows the parameters of the neurons that best approximate the functions. The table also gives the correlation coefficients between the target and actual spike frequency for each neuron. The first coefficient is for the input current sequence that was used in the parameter fitting algorithm. The second coefficient is for a test input current sequence that played no role in selecting the parameters.

The proportional neuron has a PMCC that is very nearly 1; there is a linear relationship between the input current and spike frequency. The proportional and integral neurons have a high correlation between the target and actual spike frequency for both the training current sequence and the test current sequence. The derivative neuron has a lower, but still significant, correlation for the test current sequence.

Table 1. The best results found by parameter fitting. c is the training current sequence. c_t is the test current sequence.

| Function	a	b	c	d	PMCC with c	PMCC with c_t
Proportional	0.100	0.222	-61.6	0.0	0.976	0.958
Derivative	0.0105	0.656	-55.0	1.92	0.832	0.688
Integral	0.0158	0.139	-70.0	-1.06	0.920	0.920

Fig. 3 shows the spike frequency (solid line) against the result of applying the relevant function to the input current (dashed line) of each neuron. This second metric is essentially the target spike frequency. Note that in the case of the integral function, a leaking integral of the input current is shown. Each of the three neurons is shown with the training and the test input current. For reference, the lower line in each figure shows the input current of the neuron. In all cases, the simulation was run for 2.5 seconds. The graph for the derivative neuron shows a significant amount of noise in the spike frequency. This may be solved by using a small population of neurons to represent the derivative term and using a population coding scheme.
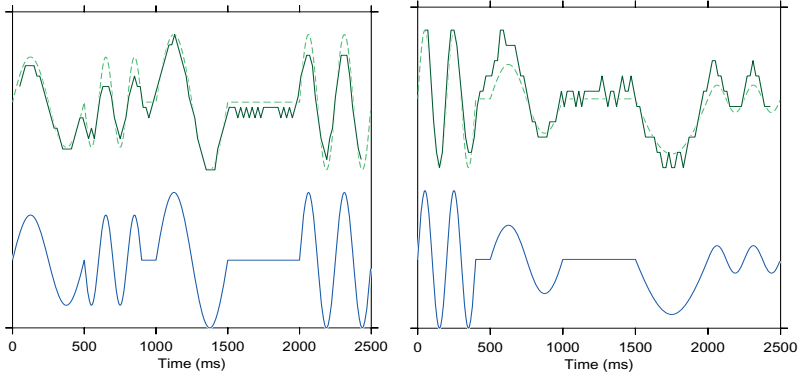
(a) Proportional neuron, training se-(b) Proportional neuron, test sequence
quence
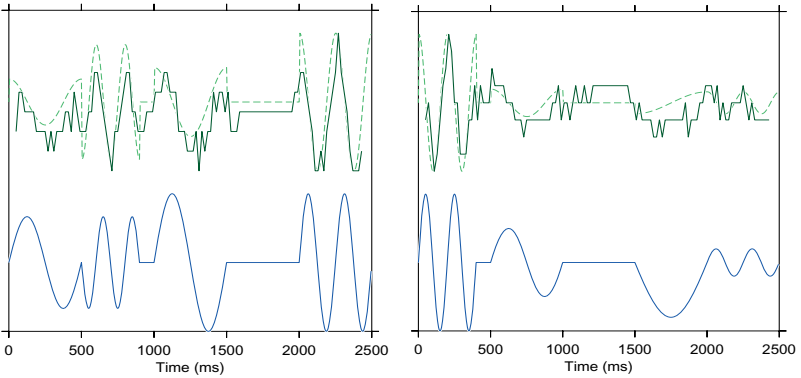
(c) Derivative neuron, training se-(d) Derivative neuron, test sequence
quence
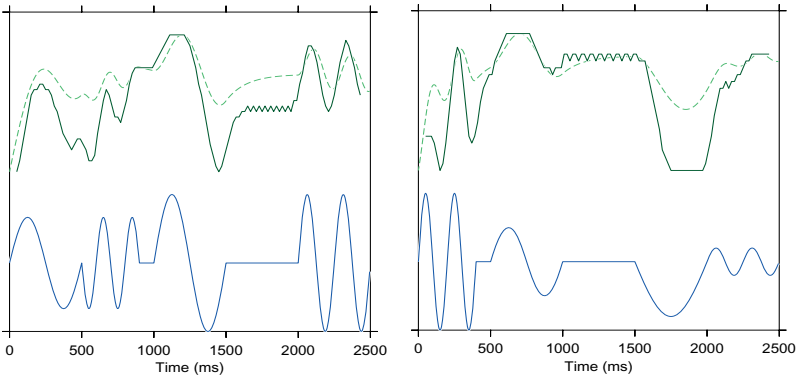
(e) Integral neuron, training sequence   (f) Integral neuron, test sequence

Fig. 3. Spike frequency (upper solid line) against target spike frequency (dashed line). The input current is also shown.

5   Conclusions

We have shown that a single biologically plausible neuron can approximate each of the terms of a PID controller, and therefore that a circuit of three neurons can act as a controller. Such a controller would have a simple design and have the modular configurability property discussed in section 1. Future work will focus on using established biological learning rules of neural networks to automatically configure and optimise a spiking neural PID controller for a simple control problem.

References

1. Dayan, P., Abbott, L.F.: Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems, 1st edn. The MIT Press (2001)
2. Furber, S., Temple, S.: Neural systems engineering. Journal of The Royal Society Interface 4(13), 193–206 (2007)
3. Gerstner, W., Kistler, W.M.: Spiking Neuron Models, 1st edn. Cambridge University Press (2002)
4. Hodgkin, A.L., Huxley, A.F.: A quantitative description of membrane current and its application to conduction and excitation in nerve. The Journal of Physiology 117(4), 500–544 (1952)
5. Izhikevich, E.M.: Simple model of spiking neurons. IEEE Transactions on Neural Networks 14(6), 1569–1572 (2003)
6. Izhikevich, E.M.: Which Model to Use for Cortical Spiking Neurons?. IEEE Transactions on Neural Networks 15(5), 1063–1070 (2004)
7. Izhikevich, E.M.: Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting (Computational Neuroscience), 1st edn. The MIT Press (2006)
8. Jin, X., Furber, S.B., Woods, J.V.: Efficient modelling of spiking neural networks on a scalable chip multiprocessor, pp. 2812–2819 (June 2008)
9. Rast, A.D., Khan, M.M., Jin, X., Plana, L.A., Furber, S.B.: A universal abstract-time platform for real-time neural networks. In: Proceedings of the 2009 International Joint Conference on Neural Networks, IJCNN 2009, pp. 3378–3385. IEEE Press, Piscataway (2009)
10. Rast, A.D., Jin, X., Galluppi, F., Plana, L.A., Patterson, C., Furber, S.: Scalable event-driven native parallel processing: the SpiNNaker neuromimetic system. In: Proceedings of the 7th ACM International Conference on Computing Frontiers, CF 2010, pp. 21–30. ACM, New York (2010)
11. Shu, H., Pi, Y.: Pid neural networks for time-delay systems. Computers & Chemical Engineering 24(2-7), 859 (2000)

Analysis on Wang's *k*WTA with Stochastic Output Nodes

John Pui-Fai Sum[1], Chi-Sing Leung[2], and Kevin Ho[3]

[1] Institute of Technology Management,
National Chung Hsing University, Taichung 402, Taiwan
`pfsum@nchu.edu.tw`
[2] Department of Electronic Engineering, City University of Hong Kong
Kowloon Tong, KLN, Hong Kong
`eeleungc@cityu.edu.hk`
[3] Department of Computer Science and Communication Engineering,
Providence University, Sha-Lu, Taiwan
`ho@pu.edu.tw`

Abstract. Recently, a Dual Neural Network-based *k*WTA has been proposed, in which the output nodes are defined as a Heaviside step activation function. In this paper, we extend this model by considering that the output nodes are stochastic. Precisely, we define this stochastic behavior by the logistic function. It is shown that the DNN-based *k*WTA with stochastic output nodes is able to converge and the convergence rates of this network are three folds. Finally, the energy function governing the dynamical behavior of the network is unveiled.

Keywords: Convergence, Dual Neural Networks, kWTA, Stochastic Output Nodes.

1 Introduction

In many classical design, a *k*WTA consists of n nodes and n^2 connections [1, 2]. By re-formulating the k winners selection problem as a linear program, Wang and his co-workers have recently proposed a Dual Neural Network (DNN) structure to implement a *k*WTA which consists of only n output nodes, and $2n$ connections [3–5]. The output nodes are defined as the Heaviside step activation function, while the hidden node behaves as a recurrent state variable[1]. This simple structure makes it suitable for hardware implementation.

However, as known in the studies of fault tolerant neural network, hardware implementation can never be perfect [7–9]. The behavior of a electronic component can always be affected by random fault. Therefore, it is inevitable to analyze the behavior of this *k*WTA if random fault exists in the output nodes [7, 10–12]. In this paper, we define the probability that the output of an output node is 1 by the logistic function.

---

[1] An identical structure has very recently been proposed independently in [6].

In the rest of the paper, the analytical results on this DNN-based kWTA with stochastic output nodes will be elucidated. In the next section, the DNN-based kWTA network will be presented. The mathematical model of a stochastic output node is defined. The convergence behavior is then analyzed in Section 3. Section 4 presents the energy function governing the dynamical behavior of the network. We conclude the paper in Section 5.

2   DNN-Based kWTA

For a general n inputs kWTA, the inputs are denoted as $u_1, u_2, \cdots, u_n$ and the outputs are denoted as $x_1, x_2, \cdots, x_n$. Without loss of generality, we assume that the values of u_is are all distinct and bounded by zero and one. Following Wang's notation, we denote $\bar{u}_n, \bar{u}_{n-1}, \cdots, \bar{u}_{n-k+1}$ as the k largest numbers, and $0 \leq \bar{u}_1 < \cdots < \bar{u}_n \leq 1$.

2.1   Deterministic Output Nodes

The DNN-based kWTA is modeled by the following state-space system[2].

$$\epsilon \frac{dy}{dt} = \sum_{i=1}^{n} x_i(t) - k, \tag{1}$$

$$x_i(t) = g(u_i - y(t)) \text{ for } i = 1, 2, \cdots, n, \tag{2}$$

where

$$g(s) = \begin{cases} 1 \text{ if } s > 0 \\ 0 \text{ otherwise.} \end{cases} \tag{3}$$

The value k in (1) is specified by the user. By (1) and (2), $y(t)$ converges[5].

2.2   Stochastic Output Nodes

In Wang's DNN-based kWTA model, the function of the output node $g(s)$ is a step function, which is a deterministic function of $(u_i - y)$, and it is implemented by threshold logics. However, the hardware components can always be affected by random noise in the circuit or by radiation. Thus, the output of the nodes turns out to be random. As a result, the output can be 1 even if $u_i \leq y$ or the output can be 0 even if $u_i > y$. We model this dynamic behavior by the following state-space system :

$$\epsilon \frac{dy}{dt} = \sum_{i=1}^{n} x_i(t) - k, \tag{4}$$

$$x_i(t) = \tilde{g}(u_i - y(t)) \text{ for } i = 1, 2, \cdots, n, \tag{5}$$

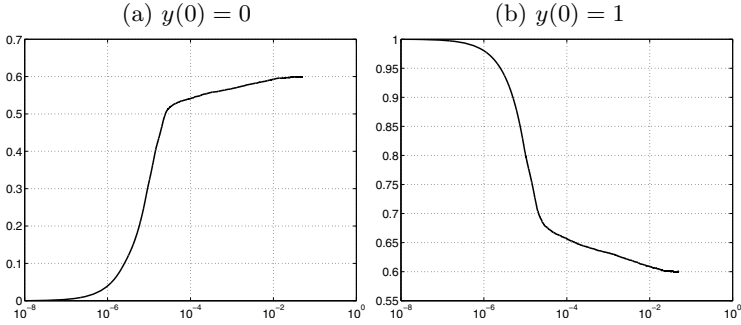[2] Equations (29) (30) in [5].

Fig. 1. Change of $y(t)$ against time. (a) $y(0) = 0$ and (b) $y(0) = 1$. Note that the scale on the time is in log scale. The inputs are 0.5, 0.7, 0.8, 0.4, 0.1 and 0.3 respectively. $k = 2$, $\epsilon = 0.0001$ and $\alpha = 100$.

where $\tilde{g}(u_i - y(t))$ is output of the ith output node at time t. To model the stochastic behavior of this output node, we define the output of $\tilde{g}(s)$ as a binary random variable. The probability that $\tilde{g}(s)$ will output 1 is determined by the logistic function :

$$P(\tilde{g}(s) = 1) = \frac{1}{1 + \exp(-\alpha s)}. \tag{6}$$

If parameter α in (6) tends to infinity, $\tilde{g}(s)$ reduces to $g(s)$.

Figure 1 shows two typical cases on the convergence of $y(t)$. The values of the input variables are $0.5, 0.7, 0.8, 0.4, 0.1$ and 0.3. The value of ϵ is 0.0001 and α is 100. It is clear that the convergence behavior of $y(t)$ and the convergence time are quite different from those obtained in the original DNN-based kWTA. For the case that $y(0) = 0$, the change of $y(t)$ goes through three phases. In the initial phase, the change of $y(t)$ is rapid. Once $y(t)$ has reached a value above 0.5, it increasing rate os proportional to $\log(t)$. When $y(t)$ reaches a value close to 0.6, it increases in a very slow rate.

3   Convergence Analysis

With the new definition on the output nodes, it is critical to investigate if the network converges. If it converges, what will be its convergence rate. To answer these questions, we need to analyze the dynamical change of $y(t)$. In this section, we will first show that $y(t)$ converges. Then, $\lim_{t\to\infty} y(t)$ will be derived. After that, we will show that $y(t)$ converges by three different rates. The energy function governing the dynamical behavior will be discussed in the next section.

Consider the change of $y(t)$ from t to $t + \tau$, we get that

$$y(t + \tau) = y(t) + \frac{1}{\epsilon} \left\{ \sum_{i=1}^{n} \int_0^\tau \tilde{g}(u_i - y(\eta))d\eta - k\tau \right\}. \tag{7}$$

By (6), the above equation (7) can be rewritten as follows :

$$y(t + \tau) = y(t) + \frac{\tau}{\epsilon} \left\{ \sum_{i=1}^{n} f(u_i - y(t)) - k \right\} \tag{8}$$

for τ is small and $f(s)$ in (8) is given by

$$f(s) = \frac{1}{1 + \exp(-\alpha s)}. \tag{9}$$

With (8), we can prove that the DNN-based kWTA with stochastic output nodes has only one equilibrium point.

Theorem 1. *For a DNN-kWTA with stochastic output nodes, there exists a unique equilibrium point. If furthermore, $|\bar{u}_{i+1} - \bar{u}_i| \geq \frac{2}{\alpha}$ for all $i = 1, \cdots, n-1$, the equilibrium point y^* is given by $y^* = (\bar{u}_{n-k} + \bar{u}_{n-k+1})/2$.*

Proof: For the uniqueness property, we let

$$F(y) = \sum_{i=1}^{n} f(u_i - y(t)) - k. \tag{10}$$

From (9), $f'(s) = \alpha f(s)(1 - f(s))$. It is thus clear that $F'(y) < 0$. In virtue of $\lim_{y \to -\infty} F(y) > 0$ and $\lim_{y \to \infty} F(y) < 0$, $F(y)$ is therefore a monotonic decreasing function with unique y^* such that $F(y^*) = 0$.

Rewrite the equation (8), we can have the following recursive equation.

$$y(t + \tau) = y(t) + \frac{\tau}{\epsilon} \left\{ \sum_{i=1}^{n-k-1} \frac{1}{1 + \exp(-\alpha(\bar{u}_i - y(t)))} \right.$$

$$+ \frac{1}{1 + \exp(-\alpha(\bar{u}_{n-k} - y(t)))} + \frac{1}{1 + \exp(-\alpha(\bar{u}_{n-k+1} - y(t)))}$$

$$\left. + \sum_{i=n-k+2}^{n} \frac{1}{1 + \exp(-\alpha(\bar{u}_i - y(t)))} - k \right\}. \tag{11}$$

By the assumption that $|\bar{u}_{i+1} - \bar{u}_i| \geq \frac{2}{\alpha}$ for all $i = 1, \cdots, n-1$, we can have the following approximations for the case when $\bar{u}_{n-k} + \frac{4}{\alpha} < y(t) < \bar{u}_{n-k+1} - \frac{4}{\alpha}$.

$$\sum_{i=1}^{n-k-1} \frac{1}{1 + \exp(-\alpha(\bar{u}_i - y(t)))} = 0. \tag{12}$$

$$\frac{1}{1 + \exp(-\alpha(\bar{u}_{n-k} - y(t)))} = \exp(\alpha(\bar{u}_{n-k} - y(t))). \tag{13}$$

$$\frac{1}{1 + \exp(-\alpha(\bar{u}_{n-k+1} - y(t)))} = 1 - \exp(-\alpha(\bar{u}_{n-k+1} - y(t))). \tag{14}$$

$$\sum_{i=n-k+2}^{n} \frac{1}{1 + \exp(-\alpha(\bar{u}_i - y(t)))} = k - 1. \tag{15}$$

By (12), (13), (14) and (15), the equation (11) can then be written as follows :

$$y(t + \tau) = y(t) + \frac{\tau}{\epsilon}\left\{\exp(\alpha(\bar{u}_{n-k} - y(t))) - \exp(-\alpha(\bar{u}_{n-k+1} - y(t)))\right\}. \tag{16}$$

For $t \to \infty$, it is clear that $\lim_{t\to\infty} y(t)$ is equal to $(\bar{u}_{n-k}+\bar{u}_{n-k+1})/2$. Hence, the proof is completed.   **Q.E.D.**

Now, we can state the convergence theorem regarding the convergence of $y(t)$ defined by (4), (5) and (6).

Theorem 2. *For a DNN-kWTA with stochastic output nodes, $y(t)$ converges. If, furthermore, $|\bar{u}_{i+1} - \bar{u}_i| \geq \frac{2}{\alpha}$ for all $i = 1, \cdots, n - 1$, $\lim_{t\to\infty} y(t) = (\bar{u}_{n-k} + \bar{u}_{n-k+1})/2$.*

Proof: For the first part of the theorem, we let y^* be the equilibrium point such that $\lim_{t\to\infty} y(t) = y^*$. So, we can have the equality that $\sum_{i=1}^{n} f(u_i - y^*) = k$. For large t, $y(t + \tau) = y^* + \Delta y(t + \tau)$ and $y(t) = y^* + \Delta y(t)$. From (8), we can get that

$$\Delta y(t + \tau) = \Delta y(t) + \frac{\tau}{\epsilon}\left\{\sum_{i=1}^{n}(f(u_i - y(t)) - f(u_i - y^*))\right\}. \tag{17}$$

Note that $\sum_{i=1}^{n} f(u_i - y)$ is a continuous function of y. By Mean Value Theorem, there exists $\xi(t) \in [y(t), y^*]$ such that

$$\sum_{i=1}^{n} f(u_i - y(t)) - f(u_i - y^*) = \sum_{i=1}^{n} f'(u_i - \xi(t))\Delta y(t). \tag{18}$$

Therefore, we can get from (17) and (18) that

$$\Delta y(t + \tau) = \left\{1 - \frac{\tau}{\epsilon}\sum_{i=1}^{n} f'(u_i - \xi(t))\right\}\Delta y(t), \tag{19}$$

where $f'(u_i - \xi(t)) = \alpha f(u_i - \xi(t))(1 - f(u_i - \xi(t)))$. Note that $0 < f'(u_i - \xi(t)) < \alpha/4$ for all $u_i, \xi(t) \in R$. As for any arbitrary values of α, n and ϵ, we can define $0 < \tau < \frac{4\epsilon}{\alpha n}$ such that $0 < \frac{\alpha\tau}{\epsilon}\sum_{i=1}^{n} f(u_i - \xi(t))(1 - f(u_i - \xi(t))) < 1$. Hence from (19), we can readily show that

$$\lim_{m\to\infty} \Delta y(t + m\tau) = \lim_{m\to\infty} \prod_{q=0}^{m}\left\{1 - \frac{\tau}{\epsilon}\sum_{i=1}^{n} f'(u_i - \xi(t + q\tau))\right\}\Delta y(t)$$

$$= 0. \tag{20}$$

In other words, $y(t)$ converges to y^*.   **Q.E.D.**

The next theorem states the convergence rates of $y(t)$ (i) if $y(t)$ has just passed the boundary of $[\bar{u}_{n-k}, \bar{u}_{n-k+1}]$ and (ii) if $y(t)$ is close to $(\bar{u}_{n-k} + \bar{u}_{n-k+1})/2$.

Theorem 3. *For $y(t) \in [\bar{u}_{n-k}, \bar{u}_{n-k+1}]$ and $|\bar{u}_{i+1} - \bar{u}_i| \geq \frac{2}{\alpha}$ for all $i = 1, \cdots, n-1$, (i) the convergence rate of $y(t)$ for $\alpha(\bar{u}_{n-k+1} - y(t)) \gg 1$ is proportional to $\alpha^{-1} \log(t)$; and (ii) the convergence rate of $y(t)$ for $\alpha(y(t) - \bar{u}_{n-k}) \gg 1$ and $\alpha(\bar{u}_{n-k+1} - y(t)) \gg 1$ is proportional to $\exp(-2\alpha t/\epsilon)$.*

Proof: Without loss of generality, we assume that $y(t)$ is increasing from zero to $\bar{u}_{n-k}$.

Case (i) Consider that $\alpha(\bar{u}_{n-k+1} - y(t)) \gg 1$, (16) can be reduced to

$$y(t + \tau) = y(t) + \frac{\tau}{\epsilon} \exp(\alpha \bar{u}_{n-k}) \exp(-\alpha y(t)). \tag{21}$$

Next, $\frac{dy}{dt}$ is approximated by $\frac{y(t+\tau) - y(t)}{\tau}$ for $\tau \ll 1$. Based on the approximation, (21) can be rewritten as follows :

$$\frac{dy}{dt} \approx \epsilon^{-1} \exp(\alpha \bar{u}_{n-k}) \exp(-\alpha y(t)). \tag{22}$$

The solution of (22) is given by[3]

$$y(t) = \alpha^{-1} \log \left(\exp(\alpha \bar{u}_{n-k}) \epsilon^{-1} \alpha t + \gamma \right), \tag{23}$$

where γ in (23) is a constant determined by the initial condition. For $t \to \infty$,

$$\begin{aligned} y(t) &\approx \alpha^{-1} \log \left(\exp(\alpha \bar{u}_{n-k}) \epsilon^{-1} \alpha \right) + \alpha^{-1} \log(t), \\ &\approx \bar{u}_{n-k} + \alpha^{-1} \log \left(\alpha \epsilon^{-1} \right) + \alpha^{-1} \log(t), \end{aligned} \tag{24}$$

For $\alpha \gg 1$, $\alpha^{-1} \log \left(\frac{\alpha}{\epsilon} \right) \to 0$. Therefore from (24), we can have

$$y(t) \approx \bar{u}_{n-k} + \alpha^{-1} \log(t). \tag{25}$$

If we further change the base of the $\log(t)$ in (25) from natural number e to 10, $y(t)$ will be given by

$$y(t) \approx \bar{u}_{n-k} + (\alpha \log_{10} e)^{-1} \log_{10}(t). \tag{26}$$

If $\alpha = 100$, $(\alpha \log_{10} e)^{-1} = 0.0230$. The value conforms with the one shown in Figure 1.

Case (ii) For $\alpha(y(t) - \bar{u}_{n-k}) \gg 1$ and $\alpha(\bar{u}_{n-k+1} - y(t)) \gg 1$, (16) can be reduced to

$$y(t + \tau) = y(t) + \frac{\tau \alpha}{\epsilon} \left\{ ((\bar{u}_{n-k} + \bar{u}_{n-k+1}) - 2y(t))) \right\}. \tag{27}$$

We can conclude from (27) that the convergence rate is proportional to $\exp(-2\alpha t/\epsilon)$. The proof is completed.   **Q.E.D.**

From the above theorem and the example given in Figure 1, the convergence of $y(t)$ if $y(0) = 0$ manifests three different phases. *(I) When $0 < y(t) < \bar{u}_{n-k}$,*

---

[3] Note that $h(t) = \alpha^{-1} \log(\beta \alpha t + \gamma)$, where γ is a constant determined by the initial condition of $h(0)$, is a solution of the differential equation $\frac{dh}{dt} = \beta \exp(-\alpha h)$.
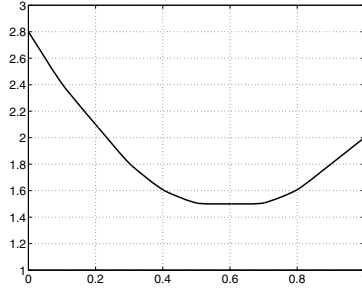
Fig. 2. Energy function being minimized, with $\alpha = 100$. The number of winners (k) is 2 and the inputs are 0.5, 0.7, 0.8, 0.4, 0.1 and 0.3 respectively.

$y(t)$ converges very fast to $\bar{u}_{n-k}$. (II) When $y(t)$ has just passed the point $\bar{u}_{n-k}$, its convergence rate is proportional to $\log(t)$. (III) When $y(t)$ is close to $(\bar{u}_{n-k} + \bar{u}_{n-k+1})/2$, its convergence rate is proportional to $\exp(-2\alpha t/\epsilon)$. Similarly, the convergence of $y(t)$ if $y(0) = 1$ manifests the same three different phases. In both cases, the longest time spanned is in the second phase. In sequel, the convergence time of a DNN-based kWTA with stochastic output nodes can be approximated by $\exp(\alpha(\bar{u}_{n-k+1} - \bar{u}_{n-k})/2)$.

4   Energy Function

It should also be noted that (8) can be rewritten as follows :

$$y(t+\tau) = y(t) - \frac{\tau}{\epsilon} \left.\frac{\partial V}{\partial y}\right|_{y=y(t)}, \tag{28}$$

where

$$V(y) = ky + \alpha^{-1} \sum_{i=1}^{n} \log\left(1 + \exp(\alpha(\bar{u}_i - y))\right). \tag{29}$$

In other word, the dynamical change of the DNN-based kWTA with stochastic output nodes (8) can be treated as a gradient descent algorithm which minimizes the energy function $V(y)$ given by (29). Figure 2 shows the shape of this energy function if the inputs are 0.5, 0.7, 0.8, 0.4, 0.1 and 0.3 and $\alpha = 100$. It is clear that $V(y)$ has just one minimum and the shape of $V(y)$ around the minimum is almost flat. It explains why the convergence rate of $y(t)$ is very small if it is close to the equilibrium point. The energy function for much larger α has also been plotted but not shown here. As expected, the shape of $V(y)$ around 0.6 is almost flat. For $\alpha \to \infty$, $V(y)$ is a piecewise linear function as derived in Wang's paper [4] and the value of $V(y)$ is constant for all $y \in (0.5, 0.7)$.

5   Conclusion

In this paper, we have presented an extended model of the DNN-based kWTA, in which the output nodes are stochastic. The probability that the node will

output 1 is governed by the logistic function. With this new definition, we have shown that the convergence of the state variable $y(t)$ is also guaranteed. However, in contrast to the original DNN-based kWTA, the convergence time of this extended model is infinite. The limit $\lim_{t\to\infty} y(t)$ could be located outside the range of $[\bar{u}_{n-k}, \bar{u}_{n-k+1}]$, especially when the inputs are too close to each other. If the inputs are sparsely located, we have shown that the convergence of $y(t)$ manifests three different values. The slowest rate appears when $y(t)$ is close to the boundary of $[\bar{u}_{n-k}, \bar{u}_{n-k+1}]$. Apart from the analyses on the convergence rate of the network, the energy function governing the dynamical behavior of the network has been unveiled.

References

1. Calvert, B.A., Marinov, C.: Another k-winners-take-all analog neural network. IEEE Transaction on Neural Networks 11(4), 829–838 (2000)
2. Majani, E., Erlanson, R., Abu-Mostafa, Y.: On the k-winner-take-all network. In: Touretzky, D. (ed.) Advances in Neural Information Processing Systems, pp. 634–642 (1989)
3. Hu, X., Wang, J.: An improved dual neural network for solving a classs of quadratic programming problems with its k-winners-take-all application. IEEE Transactions on Neural Networks 19(12), 2022–2031 (2008)
4. Wang, J., Guo, Z.: Parametric Sensitivity and Scalability of k-Winners-Take-All Networks with a Single State Variable and Infinity-Gain Activation Functions. In: Zhang, L., Kwok, J., Lu, B.L. (eds.) ISNN 2010, part I. LNCS, vol. 6063, pp. 77–85. Springer, Heidelberg (2010)
5. Wang, J.: Analysis and design of a k-Winners-take-all model with a single state variable and the Heaviside step activation function. IEEE Transactions on Neural Networks 21(9), 1496–1506 (2010)
6. Liu, Q., Dang, C., Cao, J.: A novel recurrent neural network with one neuron and finite-time convergence for k-winners-take-all operation. IEEE Transactions on Neural Networks 21(7), 1140–1148 (2010)
7. Leung, C.S., Sum, J.: A fault tolerant regularizer for RBF networks. IEEE Transactions on Neural Networks 19(3), 493–507 (2008)
8. Marinov, C.A., Costea, R.L.: Time-oriented synthesis for a WTA continuous-time neural network affected by capacitive cross-coupling. IEEE Transactions on Circuits and Systems-I: Regular Papers 57(6), 1358–1370 (2010)
9. Wang, L.: Noise injection into inputs in sparsely connected Hopfield and winner-take-all neural networks. IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics 27(5), 868–870 (1997)
10. Bolt, G.: Fault tolerant in multi-layer Perceptrons. PhD Thesis, University of York, UK (1992)
11. Judd, S., Munro, P.W.: Nets with unreliable hidden nodes learn error-correcting codes. In: Hanson, S.J., Cowan, J.D., Giles, C.L. (eds.) Advances in Neural Information Proceesing Systems, vol. 5, pp. 89–96. Morgan Kanfmanu, San Mateo (1993)
12. Sequin, C.H., Clay, R.D.: Fault tolerance in feedforward artificial neural networks. Neural Networks 4, 111–141 (1991)

Regularizer for Co-existing of Open Weight Fault and Multiplicative Weight Noise

Chi-Sing Leung[1] and John Pui-Fai Sum[2]

[1] Dept. of Electronic Engineering,
City University of Hong Kong, Hong Kong
`eeleungc@cityu.edu.hk`
[2] Institute of Electronic Commerce,
National Chung Hsing University
`pfsum@nchu.edu.tw`

Abstract. This paper first derives the training objective function of faulty radial basis function (RBF) networks, in which open weight fault and multiplicative weight noise co-exist. A regularizer is then identified from the objective function. Finally, the corresponding learning algorithm is developed. Compared to the conventional approach, our approach has a better fault tolerant ability. We then develop a faulty mean prediction error (FMPE) formula to estimate the generalization ability of faulty RBF networks. The FMPE formula helps us to understand the generalization ability of faulty networks without using a test set or generating a number of potential faulty networks. We then demonstrate how to use our FMPE formula to optimize the RBF width for the co-existing fault situation.

Keywords: RBF networks, fault tolerance.

1 Introduction

Many researchers [1–3] showed that if special care is not taken during training, the weight fault situation could lead to a drastic performance degradation. Hence, obtaining a fault tolerant neural network is very important. In the implementation of neural networks, weight faults, such as multiplicative weight noise [4] and open weight fault [5], happen unavoidably. The classical way to improve the fault tolerance is to generate a number of faulty networks during training. Injecting random weight fault [1] during training is a typical example. But the number of training epochs should be large. Simon & Sherief [6], and Zhou *et al.* [3] formulated the learning problem as an unconstrained optimization problem. The above formulations are computationally complicated when the multi-weight fault situation is considered.

Although many training methods have been developed in the past two decades, most of them focus on one kind of weight faults. For example, in [2], the algorithm was used to handle the open weight fault only. In [4], the algorithm was used to handle the multiplicative weight noise. Clearly, in the real situation, different kinds of weight faults could co-exist unavoidably.

This paper uses the RBF network model as an example to develop a regularizer for the weight fault situation in which the open weight fault and multiplicative weight noise

co-exist. We first derive an objective function for the co-existing situation. A regularizer is then identified from the objective function. With the regularizer, the training algorithm is developed. Finally, we develop a faulty mean prediction error (FMPE) formula to predict the generalization ability of faulty networks. To demonstrate the effectiveness of the FMPE formula, we discuss the way to use optimize the RBF width.

2   Background

We are given a training set: $\mathcal{D}_t = \{(\boldsymbol{x}_i, y_i) : \boldsymbol{x}_i \in \Re^K, y_i \in \Re, i = 1, \cdots, N\}$, where $\boldsymbol{x}_i$ and y_i are the input and output of the i-th sample, respectively, and K is the input dimension. The output is generated by an unknown stochastic system, given by $y_i = f(\boldsymbol{x}_i) + \epsilon_i$, where $f(\cdot)$ is a nonlinear function, and ϵ_i's are the independent zero-mean Gaussian random variables with variance σ_ϵ^2. In the RBF approach, the unknown system $f(\cdot)$ is approximated by $f(\boldsymbol{x}) \approx \hat{f}(\boldsymbol{x}, \boldsymbol{w}) = \sum_{j=1}^M w_j \phi_j(\boldsymbol{x}) = \boldsymbol{\phi}^T(\boldsymbol{x})\boldsymbol{w}$, where $\boldsymbol{w} = [w_1, \cdots, w_M]^T$ is the weight vector, $\boldsymbol{\phi}(\boldsymbol{x}) = [\phi_1(\boldsymbol{x}), \cdots, \phi_M(\boldsymbol{x})]^T$, and $\phi_j(\boldsymbol{x}) = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{c}_j\|^2}{\Delta}\right)$ is the j-th basis function. Vectors $\boldsymbol{c}_j$'s are the RBF centers. Parameter Δ controls the width of the basis functions. The training set error $\mathcal{E}(\mathcal{D}_t)$ is given by

$$\mathcal{E}(\mathcal{D}_t) = \frac{1}{N}\sum_{i=1}^N (y_i - \boldsymbol{\phi}^T(\boldsymbol{x}_i)\boldsymbol{w})^2. \tag{1}$$

Among different forms of network faults, multiplicative weight noise and open weight fault are the most common fault models [2–5, 7]. The multiplicative weight noise [4] results from the finite precision representation in the implementation of trained weights. For the open weight fault [2, 3], some RBF nodes are disconnected to the output layer.

In the real situation, the multiplicative weight noise and open weight fault could co-exist. In this situation, the mathematical model is given by

$$\tilde{w}_{j,b,\beta} = (w_j + b_j w_j)\beta_j \ \forall \ j = 1, \cdots, M, \tag{2}$$

where b_j's are identical independent zero-mean random variables with variance σ_b^2, the density function of b_j's are symmetric. In this way, the weight noise $b_j w_j$ is proportional to the magnitude of the weight. For weight fault, the fault factors β_j's are identical independent binary random variables that describes whether the j-th weight operates properly. The j-th weight is out of work when $\beta_j = 0$, otherwise it operates properly. The probability mass function of the fault factor is given by $\text{Prob}(\beta_j = 0) = p$ and $\text{Prob}(\beta_j = 1) = 1 - p$.

In this model, if a weight is not damaged (not opened), the implemented weight is affected by the multiplicative weight noise (precision error) only. On the other hand, when the weight is with open weight fault, the faulty weight are clamped at zero. In the vector notation, the faulty weight vector is given by

$$\tilde{\boldsymbol{w}}_{b,\beta} = (\boldsymbol{w} + \boldsymbol{b} \otimes \boldsymbol{w}) \otimes \boldsymbol{\beta} = \boldsymbol{w} \otimes \boldsymbol{\beta} + \boldsymbol{b} \otimes \boldsymbol{w} \otimes \boldsymbol{\beta} \tag{3}$$

where $\tilde{\boldsymbol{w}}_{b,\beta} = [\tilde{w}_{1,b,\beta}, \cdots, \tilde{w}_{M,b,\beta}]^T$, $\boldsymbol{b} = [b_1, \cdots, b_M]^T$, $\boldsymbol{\beta} = [\beta_1, \cdots, \beta_M]^T$, and $\otimes$ is the element-wise multiplication operator.

3   Objective Function, Regularizer, and Training Algorithm

Given a faulty weight vector $\tilde{\boldsymbol{w}}_{b,\beta}$, from (2) and (3), the training error of a faulty network is given by

$$
\mathcal{E}(\mathcal{D}_t)_{b,\beta} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \boldsymbol{\phi}^T(\boldsymbol{x}_i)\tilde{\boldsymbol{w}}_{b,\beta})^2
$$

$$
=\frac{1}{N}\sum_{i=1}^{N}\left[y_i^2 - 2y_i\sum_{j=1}^{M}\beta_j w_j\phi_j(\boldsymbol{x}_i) + \sum_{j=1}^{M}\sum_{j'=1}^{M}\beta_j\beta_{j'}w_jw_{j'}(1+b_jb_{j'})\phi_j(\boldsymbol{x}_i)\phi_{j'}(\boldsymbol{x}_i)\right.
$$

$$
\left.+\sum_{j=1}^{M}\sum_{j'=1}^{M}(b_j+b_{j'})\beta_j\beta_{j'}w_jw_{j'}\phi_j(\boldsymbol{x}_i)\phi_{j'}(\boldsymbol{x}_i) - 2y_i\sum_{j=1}^{M}b_j\beta_jw_j\phi_j(\boldsymbol{x}_i)\right]. \tag{4}
$$

According to the definition of the weight fault, we have $\langle\beta_i\rangle = \langle\beta_i^2\rangle = 1 - p$, and $\langle\beta_i\beta_{i'}\rangle = (1-p)^2\,\forall\,i \neq i'$, where $\langle\cdot\rangle$ is the expectation operator. For weight noise, we have $\langle b_i\rangle = 0$, $\langle b_i^2\rangle = \sigma_b^2$, and $\langle b_ib_{i'}\rangle = 0\,\forall\,i \neq i'$. Taking the expectation over b's and β's, the training error of a faulty network is given by

$$
\bar{\mathcal{E}}(\mathcal{D}_t)_{b,\beta} = \langle\mathcal{E}(\mathcal{D}_t)_{b,\beta}\rangle_{b,\beta}
$$

$$
= \frac{1}{N}\sum_{i=1}^{N}\left[y_i^2 - 2(1-p)y_i\sum_{j=1}^{M}w_j\phi_j(\boldsymbol{x}_i) + [(1-p)^2\sum_{j=1}^{M}\sum_{j'=1}^{M}w_jw_{j'}\phi_j(\boldsymbol{x}_i)\phi_{j'}(\boldsymbol{x}_i)\right.
$$

$$
\left.+[p-p^2+\sigma_b^2(1-p)]\sum_{j=1}^{M}w_j^2\phi_j^2(\boldsymbol{x}_i)\right]
$$

$$
= \frac{1-p}{N}\sum_{i=1}^{N}(y_i - \boldsymbol{\phi}^T(\boldsymbol{x}_i)\boldsymbol{w})^2 + \frac{p}{N}\sum_{i=1}^{N}y_i^2 + (1-p)\boldsymbol{w}^T[(p+\sigma_b^2)\boldsymbol{G} - p\boldsymbol{H}]\boldsymbol{w}, \tag{5}
$$

where $\boldsymbol{H} = \frac{1}{N}\sum_{j=1}^{N}\boldsymbol{\phi}(\boldsymbol{x}_i)\boldsymbol{\phi}^T(\boldsymbol{x}_i)$, and $\boldsymbol{G} = \mathbf{diag}(\boldsymbol{H})$. Equation (5) tells us the training error of faulty RBF networks. Since the term $\frac{p}{N}\sum_{j=1}^{N}y_i^2$ in (5) is independent of $\boldsymbol{w}$, minimizing the training error of faulty networks, i.e.(5), is equivalent to minimizing *the following objective function*:

$$
\mathcal{L}(\boldsymbol{w}) = \frac{1}{N}\sum_{i=1}^{N}(y_i - \boldsymbol{\phi}^T(\boldsymbol{x}_i)\boldsymbol{w})^2 + \boldsymbol{w}^T[(p+\sigma_b^2)\boldsymbol{G} - p\boldsymbol{H}]\boldsymbol{w}. \tag{6}
$$

In (6), the second term is similar to the conventional regularization term in regularization techniques [8, 9]. Hence, we could define the *co-exist fault regularizer* as

$$
\boldsymbol{w}^T[(p+\sigma_b^2)\boldsymbol{G} - p\boldsymbol{H}]\boldsymbol{w}. \tag{7}
$$

where $(p+\sigma_b^2)\boldsymbol{G}-p\boldsymbol{H}$ is the so-called regularization matrix. Hence, the optimal weight vector for minimizing the training error under co-existing fault is given by

$$
\boldsymbol{w} = \left((1-p)\boldsymbol{H} + (p+\sigma_b^2)\boldsymbol{G}\right)^{-1}\frac{1}{N}\sum_{i=1}^{N}\boldsymbol{\phi}(\boldsymbol{x}_i)y_i. \tag{8}
$$

In (6), if the open weight fault probability is large (p is large), some eigenvalues of the regularizer matrix are positive and some eigenvalues are negative. That means, for attaining the co-existing fault tolerance, the regularization matrix should contain both positive and negative eigenvalues.

4   Faulty Mean Prediction Error of Faulty Networks

In (5), we know the training error of faulty networks. Besides, with (8) we know the way to minimize the training error of faulty networks. However, in many situations, we would like to know how well the network performs on unseen samples. This section will derive a way to estimate the generalization ability for faulty networks trained with the co-existing fault regularizer.

Recall that $\mathcal{D}_t = \{(\boldsymbol{x}_j, y_j)\}_{j=1}^{N}$ is the training set. The training error of faulty networks can be rewritten as

$$\bar{\mathcal{E}}(\mathcal{D}_t)_{b,\beta} = \left\langle (y - \boldsymbol{\phi}^T(\boldsymbol{x})\tilde{\boldsymbol{w}}_{b,\beta})^2 \right\rangle_{\mathcal{D}_t, b, \beta}$$
$$= \langle y^2 \rangle_{\mathcal{D}_t} - 2(1-p)\langle y\boldsymbol{\phi}^T(\boldsymbol{x})\boldsymbol{w}\rangle_{\mathcal{D}_t} + (1-p)^2 \boldsymbol{w}^T \boldsymbol{H} \boldsymbol{w} + (1-p)(p+\sigma_b^2)\boldsymbol{w}^T \boldsymbol{G} \boldsymbol{w}. \quad (9)$$

Let $\mathcal{D}_f = \{(\boldsymbol{x}_i', y_i')\}_{i=1}^{N'}$ be the testing data set. Similarly, the test error of faulty networks is given by

$$\bar{\mathcal{E}}(\mathcal{D}_f)_{b,\beta} = \left\langle (y' - \boldsymbol{\phi}^T(\boldsymbol{x}')\tilde{\boldsymbol{w}}_{b,\beta})^2 \right\rangle_{\mathcal{D}_f, b, \beta}$$
$$= \langle y'^2 \rangle_{\mathcal{D}_f} - 2(1-p)\langle y'\boldsymbol{\phi}^T(\boldsymbol{x}')\boldsymbol{w}\rangle_{\mathcal{D}_f} + (1-p)^2 \boldsymbol{w}^T \boldsymbol{H}' \boldsymbol{w} + (1-p)(p+\sigma_b^2)\boldsymbol{w}^T \boldsymbol{G}' \boldsymbol{w} \quad (10)$$

where $\boldsymbol{H}' = \frac{1}{N'}\sum_{i=1}^{N'} \boldsymbol{\phi}(\boldsymbol{x}_i')\boldsymbol{\phi}^T(\boldsymbol{x}_i')$ and $\boldsymbol{G}' = \mathbf{diag}(\boldsymbol{H}')$. Denote the true weight vector as $\boldsymbol{w}_o$. Hence,

$$y_i = \boldsymbol{\phi}^T(\boldsymbol{x}_i)\boldsymbol{w}_o + \epsilon_i \quad \text{and} \quad y_i' = \boldsymbol{\phi}^T(\boldsymbol{x}_i')\boldsymbol{w}_o + \epsilon_i', \quad (11)$$

where ϵ_i's and ϵ_i''s are independent zero-mean Gaussian random variables with variance σ_ϵ^2. Since $\boldsymbol{w}$ is obtained entirely from $\mathcal{D}_t$, the second term in (10) can be expressed as

$$-2(1-p)\langle y'\boldsymbol{\phi}^T(\boldsymbol{x}')\boldsymbol{w}\rangle_{\mathcal{D}_f} = -2(1-p)(\frac{1}{N'}\sum_{i=1}^{N'} y_i'\boldsymbol{\phi}^T(\boldsymbol{x}_i'))\boldsymbol{w}. \quad (12)$$

From (8) and (11), (12) is then given by

$$-2(1-p)(\frac{1}{N'}\sum_{i=1}^{N'} y_i'\boldsymbol{\phi}^T(\boldsymbol{x}_i'))[(1-p)\boldsymbol{H} + (p+\sigma_b^2)\boldsymbol{G}]^{-1}\frac{1}{N}\sum_{i=1}^{N} y_i\boldsymbol{\phi}(\boldsymbol{x}_i). \quad (13)$$

As ϵ_i's and ϵ_i''s are independent, from (11), "$\frac{1}{N'}\sum_{i=1}^{N'} y_i'\boldsymbol{\phi}^T(\boldsymbol{x}_i')$" and "$\frac{1}{N}\sum_{i=1}^{N} y_i\boldsymbol{\phi}(\boldsymbol{x}_i)$" are equal to

$$\frac{1}{N'}\sum_{i=1}^{N'} y_i'\boldsymbol{\phi}^T(\boldsymbol{x}_i') = \boldsymbol{w}_o^T \boldsymbol{H}' \quad \text{and} \quad \frac{1}{N}\sum_{i=1}^{N} y_i\boldsymbol{\phi}(\boldsymbol{x}_i) = \boldsymbol{H}\boldsymbol{w}_o. \quad (14)$$

From (14), the second term in (10) becomes

$$-2(1-p)\langle y'\phi^T(x')w\rangle_{\mathcal{D}_f} = -2(1-p)w_o^T H'[(1-p)H+(p+\sigma_b^2)G]^{-1}Hw_o. \quad (15)$$

Using the similar method, the second term in (9) can be expressed as

$$-2(1-p)\langle y\phi^T(x)w\rangle_{\mathcal{D}_t} =$$
$$-2(1-p)\left(w_o^T H[(1-p)H+(p+\sigma_b^2)G]^{-1}Hw_o+\frac{\sigma_\epsilon^2}{N}\mathrm{Tr}\left\{[(1-p)H+(p+\sigma_b^2)G]^{-1}H\right\}\right), \quad (16)$$

where $\mathrm{Tr}\{\cdot\}$ denotes the trace operation. Following the common practice, for large N and N', we can assume that $H' \approx H$, $G' \approx G$, and $\langle y'^2\rangle_{\mathcal{D}_f} \approx \langle y^2\rangle_{\mathcal{D}_t}$. The difference between the generalization error of faulty networks and the training error of faulty networks is given by

$$\bar{\mathcal{E}}(\mathcal{D}_f)_{b,\beta} - \bar{\mathcal{E}}(\mathcal{D}_t)_{b,\beta} = 2(1-p)\frac{\sigma_\epsilon^2}{N}\mathrm{Tr}\left\{[(1-p)H+(p+\sigma_b^2)G]^{-1}H\right\}. \quad (17)$$

From (9) and (17), the generalization error of faulty networks becomes

$$\bar{\mathcal{E}}(\mathcal{D}_f)_{b,\beta} = (1-p)\mathcal{E}(\mathcal{D}_t)+\frac{p}{N}\sum_{i=1}^{N} y_i^2 + (p^2-p)w^T Hw + (1-p)(p+\sigma_b^2)w^T Gw$$
$$+2(1-p)\frac{\sigma_\epsilon^2}{N}\mathrm{Tr}\left\{[(1-p)H+(p+\sigma_b^2)G]^{-1}H\right\}. \quad (18)$$

In (18), the first term $(1-p)\mathcal{E}(\mathcal{D}_t)$ is the training set MSE of the trained fault-free network. It can be directly obtained after training. The second term $\frac{p}{N}\sum_{i=1}^{N} y_i^2$ can be obtained from the training set. The last two terms are in terms of matrices H and G. These two matrices can be obtained from the training set too. The only unknown is the variance σ_ϵ^2 of the measurement noise. The measurement noise can be estimated based on the Fedorov's method [10], given by

$$\sigma_\epsilon^2 \approx \frac{1}{N-M}\sum_{i=1}^{N}(y_i-\phi^T(x_i)H^{-1}\frac{1}{N}\sum_{i'=1}^{N}\phi(x_{i'})y_{i'})^2. \quad (19)$$

5 Simulations

5.1 Fault Tolerance

To verify our theoretical results, we consider two data sets: (i) the sinc function and (ii) a nonlinear autoregressive time series (NAR) [11]. The sinc function is a common benchmark example. The output is generated by $y = \mathrm{sinc}(x) + \epsilon$, where the measurement noise ϵ is a zero-mean Gaussian random variable with variance $\sigma_\epsilon^2 = 0.01$. The input x is randomly taken from -5 to 5. Both training set and test set contain 200 samples. The network model has 37 RBF nodes. The RBF centers are uniformly selected from the range $[-5, 5]$. The RBF width Δ is set to 0.1.
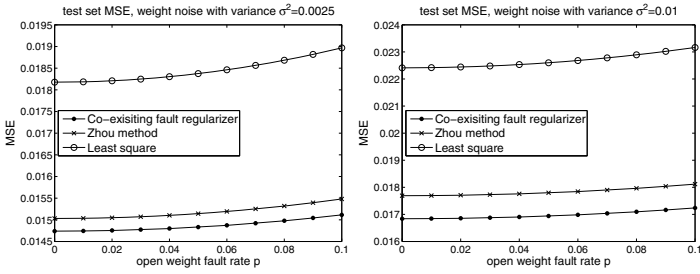
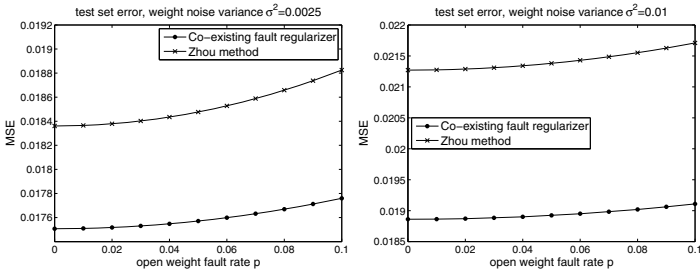**Fig. 1.** MSE of faulty networks for Sinc function example

Fig. 2. MSE of faulty networks for the dynamical system. Since the MSEs of the least square method are too high, we do not plot it in the figure.
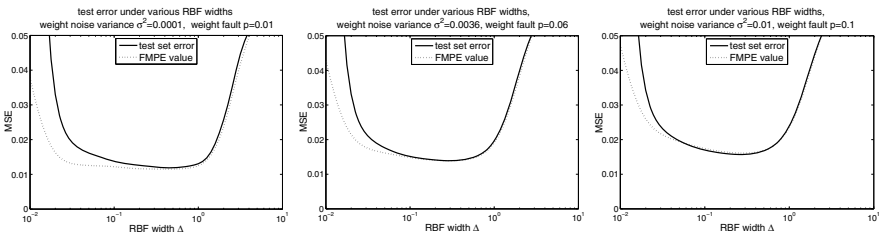

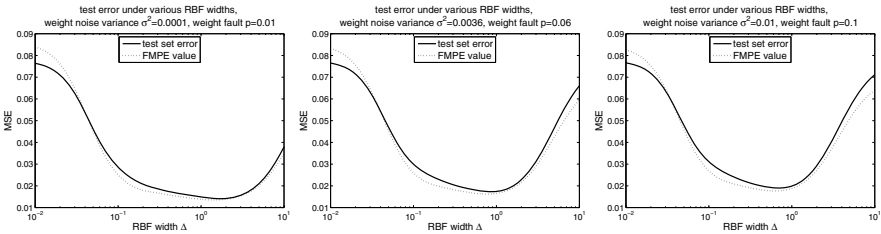**Fig. 3.** Selection of RBF width from the FMPE approach for Sinc function example

Fig. 4. Selection of RBF width from the FMPE approach for the dynamical system

We also consider a nonlinear dynamical system [11], given by

$$y(t+1) = \frac{y(t)y(t-1)y(t-2)(y(t-2)-1)x(t-1)+x(t)}{1+y^2(t-1)+y^2(t-2)} + \epsilon(t+1), \quad (20)$$

where $x(t)$ is the input, and the noise term $\epsilon(t)$ is a zero mean Gaussian variable with variance $\sigma_\epsilon^2 = 0.1^2$. We generate 400 samples with $y(0) = y(-1) = y(-2) = 0$. To excite all the modes of the system, inputs $x(t)$'s are random signal generated in the range of $[-0.5, 0.5]$. The first 200 samples are used for training and other samples are used as test set. The network model has 50 RBF nodes. The RBF centers are randomly selected from the training set. In addition, the RBF width Δ is set to 0.6.

Two other techniques are also considered in the simulation. They are the pseudo inverse and Zhou's method [3]. The least square is a reference which tests the performance of faulty networks when special care is not considered. The average test MSEs, under various weight noise levels and open weight fault levels, are shown in Figures 1 and 2. In Figure 2, **we do not show the MSEs of the least square method because its MSE values are too high (greater than 10)**. From our result, the least method has very poor performance. This result confirms that without special care during the performance of faulty networks could be very poor. The Zhou's method and our co-existing fault regularizer can improve fault tolerance. Compared with the Zhou's method, our approach has a better performance. The reason is that the co-existing fault regularizer aims at minimizing the MSE of faulty networks.

5.2   Application of FMPE Formula: Selecting RBF Width

The FMPE formula help us not only to estimate generalization ability of a trained network but also to select some model parameters. In the RBF approach, one tuning parameter is the RBF width Δ. We illustrate how our MPE results can help us to select an appropriate value of Δ. Following the conventional approaches in selecting parameters for fault–free networks, we try different values of Δ. Afterwards, we use the FMPE formulae to estimate the test error of faulty networks. The results are depicted in Figure 3 and Figure 4.

From the figures, although there are small differences between the true test errors [1] and FMPE values, our method can locate optimal Δ for minimizing the generalization error of faulty networks.

For example, for the dynamic system example with weight noise variance $\sigma_b^2 = 0.0001$ and open weight fault rate $p = 0.01$, the searched Δ is 1.413 and the corresponding test set error is 0.01421. When we use the brute force way (test set method) to search Δ, the test set error is 0.01409. With weight noise variance $\sigma_b^2 = 0.01$ and open weight fault rate $p = 0.1$, the searched Δ is 0.631 and the corresponding test set error is 0.01905. When we use the brute force way (test set method) to search Δ, the test set error is 0.01899. The simulation confirms the applicability of our FMPE results for the selection of RBF width. For other faulty levels and examples, we obtained similar results (not shown here).

---

[1] When we use the test set method, we need to have a test set and generate a number of faulty networks to measure the performance of faulty networks under different weight noise and weight fault levels.

6   Conclusion

This paper addresses the fault tolerance of RBF networks when the weight noise and weight fault happen co-exists. An objective function for minimizing the training error of faulty networks are then derived. A regularizer for handling the co-existing fault situation is identified from the objective function. The corresponding learning algorithm is developed. In our approach, the objective function and the learning algorithm are computationally simple. Besides, we also derive the a formula to predict the generalization performance of the faulty networks trained from our algorithm. In the simulation, we demonstrate the way to use the formula to optimize the RBF width.

References

1. Chiu, C.T., Mehrotra, K., Mohan, C.K., Ranka, S.: Modifying training algorithms for improved fault tolerance. In: Proceedings of the International Conference on Neural Networks 1994, vol. 4, pp. 333–338 (1994)
2. Leung, C.S., Sum, J.: A fault-tolerant regularizer for RBF networks. IEEE Trans. Neural Netw. 19(3), 493–507 (2008)
3. Zhou, Z.H., Chen, S.F.: Evolving fault-tolerant neural networks. Neural Computing and Applications 11(3-4), 156–160 (2003)
4. Burr, J.: Digital neural network implementations. In: Neural Networks, Concepts, Applications, and Implementations, vol. III, pp. 237–285. Prentice-Hall, Englewood Cliffs (1995)
5. Bolt, G.: Fault Tolerant Multi-Layer Perceptron Networks. PhD thesis, Department of Computer Science, University of York, UK (July 1992)
6. Simon, D., El-Sherief, H.: Fault-tolerance training for optimal interpolative nets. IEEE Trans. on Neural Netw. 6, 1531–1535 (1995)
7. Phatak, D.S., Tchernev, E.: Synthesis of fault tolerant neural networks. In: Proc. IJCNN 2002, pp. 1475–1480 (2002)
8. Moody, J.E.: Note on generalization, regularization, and architecture selection in nonlinear learning systems. In: Proc. First IEEE-SP Workshop on Neural Networks for Signal Processing, pp. 1–10 (1991)
9. Leung, C.S., Tsoi, A.C., Chan, L.W.: Two regularizers for recursive least square algorithms in feedforward multilayered neural networks. IEEE Trans. Neural Netw. 12(6), 1314–1332 (2001)
10. Fedorov, V.V.: Theory of optimal experiments. Academic Press (1972)
11. Narendra, K.S., Parthasarathy, K.: Neural networks and dynamical systems. International Journal of Approximate Reasoning 6(2), 109–131 (1992)

Research on a RBF Neural Network in Stereo Matching

Sheng Xu, Ning Ye, Fa Zhu, Shanshan Xu, and Liuliu Zhou

College of Information Science and Technology,
Nanjing Forestry University, Nanjing 210037, China

Abstract. There are so many shortcomings in current stereo matching algorithms, for example, they have a low robustness, so as to be influenced by the environment easily, especially the intensity of the light and the number of the occlusion areas; also they often have a poor practical performance for they are difficult to deal with the matching problem without knowing the disparity range and have a high complexity when using the global optimization. In order to solve the above problems, here design a new stereo matching algorithm called RBFSM which main uses the RBF neural network (RBFNN). The RBFSM will get the correspondence between the input layer nodes and hidden layer nodes by the Gaussian function and then use the weight matrix between the hidden layer and output layer to calculate input pixels' disparity. Here will give the analysis of this new RBF neural network matching algorithm through a lot of experiments, and results show that the new algorithm not only overcome the shortcomings of the traditional methods like low robustness and low practical performance, but also can improve the matching precision significantly with a low complexity.

Keywords: Robustness, Practical performance, RBFNN, Correspondence, Stereo matching.

1 Introduction

With two identical cameras shooting the same scenario from two angles, we can get two pictures. According to these two photographs, we will find a corresponding relationship between left image pixels and right image pixels. Using this mapping we can also get the objects actual space depth information in the scene and solving this correspondence problem is called the stereo matching.

Current stereo matching algorithms are based on regional or feature [1]. These algorithms usually convert the problem into minimizing an energy equation, and to solve the energy equation means to get the disparity [2]. Here the disparity is the answer of the stereo matching problem. There are so many algorithms, for example, SAD, SO (Scanline Optimization) [3], DP (Dynamic Programming) [4]. However the above algorithms are belong to local optimization methods. While later the GC (Graph cut) [5] and BP (Belief Propagation) [6] deal with the matching problem by global energy optimization methods so as to get a much better result, All these above algorithms need to know the disparity search range and cannot avoid the redundant search. Now the below algorithms give a different way to deal with the stereo matching, and the basic theory is the neural network [7]. For example, the Hopfield neural network [8], and a relatively good method MSOM [9] which uses the

self-organizing neural network to do the stereo matching. Taking into account all the advantages of neural network algorithms and knowing the Gaussian radial basis function network can preserve a good pixel relationship between the input layer nodes and the hidden layer nodes. Here design a new algorithm called RBFSM which innovative uses of the RBF neural network.

Compared with the above neural network methods, the RBFSM shows a good robustness. It still works well even when the two matching images have a quite different light intensity or one of the images is rotated. Both these two advantages are not met by the above Hopfield network or self-organization network. Because of the good pixel relationship, it can give a higher precision in the occlusion area than the current methods. Furthermore it has a good practical performance that it does not need to know the disparity range which methods do not use the neural network are impossible to have like SAD, SO and DP, etc. The RBFSM has only one hidden layer and the network structure is not complex so it can improve the precision significantly while with a low complexity and the high complexity is exactly the major disadvantage of the global optimization methods.

2   The RBFSM Algorithm

2.1   The RBF Neural Network

The output of the last layer in RBFNN can be described by the following (1).

$$Y_i = \sum_{c=1}^{n} \omega_{ic} Hid_c(X) \tag{1}$$

The input sample is *X*, and the hidden layer function is **Hid**. Here *c* indicates the order of hidden nodes, and there are **n** nodes in the hidden layer. Through the **Hid**, the input will be mapped to the hidden layer, reflecting the information between hidden nodes and input nodes. Next, let the matrix which is output from the hidden layer multiply the weight ω, here *i* is the order of the last layer nodes, and then we will get *i*-th node output Y_i in this layer.

2.2   The Design of the RBFSM Algorithm

Obviously, according to the traditional RBF neural network, we are unable to solve the stereo matching problems. It should be made some changes to make the input *X* is the left pixels and when get the output *Y*, we will also obtain the input pixels' disparity.

Steps of the RBFSM algorithm are as follows:

1. Note a pixel in the right image $P_R(M, N)$, where *M* is the horizontal axis and *N* is the vertical axis for the point P_R, then define three attributes for P_R, they are $P_R(M,N)_i$, $P_R(M,N)_j$ and $P_R(M,N)_I$, as (2)~(4)

$$P_R(M, N)_i = M \tag{2}$$

$$P_R(M, N)_j = N \tag{3}$$

$$P_R(M, N)_I = I_R(M, N) \tag{4}$$

The $I_R(M,N)$ expressed the intensity of the pixel whose axis is (M,N) in the right image, and then we selected two special regions for the corresponding point P_R. As shown in Fig. 1, the rectangular area B_R1 which is K_{in} pixels from the center point P_R, and the rectangular area B_R2 which is K_{out} pixels from the center point P_R.

2. From the left image, the algorithm should look for a B_L2 to be best match for the B_R2, which means to find a region with the same size in the left, and make the two regions have the minimum distance between their pixels. The way to find the B_L2 and its center point is $P_L(p,q)$ is (5):

$$(p,q) = \min_{y=N,N+1,N+2,...,W,\, x=M\le H} (B_L 2(x,y) - B_R 2(M,N)) \tag{5}$$

Here we define the minimum distance is (6). The distance includes the intensity and axis of the two areas pixels.

$$B_L 2(x,y) - B_R 2(m,n) = \sum_{j=-K_{out}}^{K_{out}} \sum_{i=-K_{out}}^{K_{out}} (P_L(p+j,q+i) - P_R(m+j,n+i))^2$$

$$= \sum_{j=-K_{out}}^{K_{out}} \sum_{i=-K_{out}}^{K_{out}} ((I_L(x+j,y+i) - I_R(m+j,n+i))^2 + (x-m)^2 + (y-n)^2) \tag{6}$$

Note that the vertical disparity is assumed to be 0, that is, the algorithm only searches in one line, so in (5) the x is m while the y ranges from N to W. The W is the width of the image while the H is the height. At this point we get the center of the regional B_L2 and that is the $P_L(p,q)$, see Fig. 2. Now the B_L2 is the best block region to match the B_R2, with the help of center point P_L, we can find the B_L1. The RBFSM has a hidden layer with 5 nodes. There are the center P_L and four corners of the B_L1 region, as shown in Fig. 2
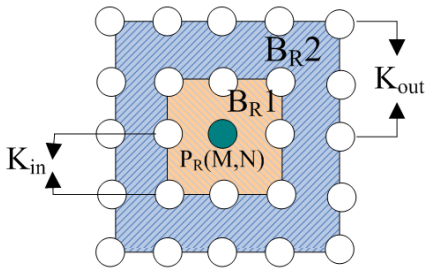

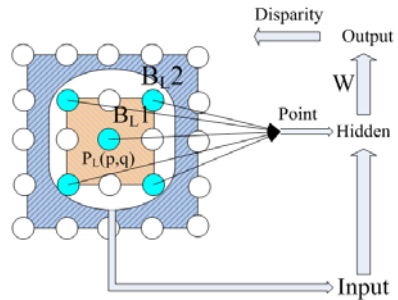**Fig. 1.** The block to be matched in the right image

Fig. 2. How the block pixels in the left image get their disparity

The five points are in the Point set. See (7), and the input X is the whole pixels in B_L1 region as the (8) expresses.

$$Point = \{P_L(p+j,q+i), P_L(p,q) / j = -K_{in}, K_{in}, i = -K_{in}, K_{in}\} \tag{7}$$

$$X = \{P_L(p-j,q-i) / j = -K_{in}, -K_{in}+1,..., K_{in}; i = -K_{in}, -k_{in}+1,..., K_{in}\} \tag{8}$$

The benefit of only considering 5 nodes is that it can handle the matching block. Not only can improve the accuracy but also speed up the efficiency of the algorithm. Only considering the part of pixels is due to the pixels far away from the center will be given a very smaller impact. Therefore, it eliminates the need for the calculation of the distance pixels. We get the hidden layer output by (9). In (9) the $hidden_k$ expresses the *k-th* center hidden node, while the output matrix of the hidden layer nodes is the *hidden* (10). Each column of the *hidden* represents the affection of a hidden node makes on *X*.

$$hidden_k = G(\| X - Point_k \|^2) = \exp(\frac{\| X - Point_k \|^2}{-2\sigma^2}) \quad k \leq 5 \tag{9}$$

$$hidden = (hidden_1, hidden_2, hidden_3, hidden_4, hidden_5) \tag{10}$$

The weights matrix ω between the hidden layer and output layer saves the displacement from the left to the right image. Using the correspondence relationship from the hidden layer, we can calculate the disparity for the whole B_L1 region. Specific method is to calculate the offset from the center of B_L2 to the center of B_R2, and the horizontal direction is (11).

$$Hor = q - N \tag{11}$$

Then calculate the location difference between the pixels in right which have the same offset as the center point $P_R(M,N)$ and the corresponding pixels in the left image. Then each pixel $P_L(x,y)$ in left can get its horizontal disparity $matrix\_Hor(x,y)_i$ in (12), and the η is the learning efficiency. The Steps of the way to get the disparity can reference to the Fig. 2. That is starting from the input layer, through the hidden layer, and then together with the weight ω to arrive at the output layer, at the last the disparity of the input will be shown by the last layer.

$$matrix\_Hor(x,y)_i = \eta \times \omega \times hidden$$
$$p - K_{in} \leq x \leq p + K_{in}, q - K_{in} \leq y \leq q + K_{in} \tag{12}$$

The matrix ω which is between the hidden layer and output layer is shown in (13).

$$\omega = P_R(x,y)_i - Hor - P_L(x,y)_i \tag{13}$$

$$temp\_Hor(x,y)_i = f(matrix\_Hor(x,y)_i) \tag{14}$$

$$f(F) = f(F_1, F_2, F_3, ..., F_n) = (F_1 + F_2 + F_3 + ... + F_n)/n \tag{15}$$

When we try to get the $temp\_Hor(x,y)_i$, we should consider the impact of all the hidden nodes, so we use a function *f* to achieve the purpose ,see (14). Often we use an easy method, which is just calculating the mean of the vector in (15)

At this time we have obtained the disparity for each pixel within B_L1. Then do the similar process to other pixels in the left image, and pixels to be processed can be in ordered or can be random. Note that before the iteration we should update the coordinates for B_L1 pixels by (16)

$$P_L(x,y)_i = P_L(x,y)_i + temp\_Hor(x,y)_i$$
$$p - K_{in} \leq x \leq p + K_{in} \quad q - K_{in} \leq y \leq q + K_{in} \tag{16}$$

Iterate the steps above. When all pixels are calculated by iteration at least once, the left image will be transferred into a new image. Then, let the coordinates of each pixel

in the new image minus the initial pixels' coordinate in the left image, and that is the horizontal disparity of each pixel in left image as the (17) shows in ***disMap_hor(x,y)***$_i$.

$$disMap\_hor(x, y)_i = y - P_L(x, y)_i \quad 1 \leq x \leq H \quad 1 \leq y \leq W \tag{17}$$

3   Implementation and Results

3.1   Algorithms Comparison and Parameters Setting

Now we will compare the RBFSM with some typical stereo matching algorithms, including SAD, SO, DP, GC, BP, and the MSOM, to prove the superiority of the RBFSM algorithm.

The images for the experiments come from the well known website: ***http://vision.middlebury.edu***. There are Tsukuba, Venus, Cones, and Teddy. All of them have been calibrated and the disparity in the vertical direction is **0**. To further reflect the superiority of the new algorithm, the RBFSM not only consider the above images in their standard situation, but also consider these images in non-standard situation that current stereo matching methods are very difficult to solve, including the high brightness, and the rotation.

Now let us determine the parameters. Considering the Tsukuba first, through a large number of experiments, we find σ is 20 and the learning efficiency $\boldsymbol{\eta}$ is 0.005 will get the best disparity map. Here we make the σ as a constant instead of a varied in the traditional RBFNN. The $\boldsymbol{K_{out}}$. is mainly used to find the corresponding between the right image point $\boldsymbol{P_R}$ and the left image point $\boldsymbol{P_L}$. while the $\boldsymbol{K_{in}}$ is mainly to construct the $\boldsymbol{B_R 1}$ region, deciding the pixels to enter the net. Here we know when the $\boldsymbol{K_{out}}$ is 3 and $\boldsymbol{K_{in}}$ is 1, the result seems to be the best. Now parameters have been set, but remember these parameters are just suitable for the Tsukuba. So the different images should be given different parameters.

3.2   Standard Situation

Here we show the style of stereo matching images for Tsukuba in Fig. 3, the a) is the left image, b) is the right image and the c) is the groundtruth.

Test the RBFSM algorithm in different images, including the Tsukuba, Venus, Cones and Teddy. We give three disparity maps in Fig. 4 from a) to c), and their groundtruth maps are from d) to f). As shown in the Fig. 4, the disparity map appears smooth and there is no ladder in it. This is because it is different from the traditional algorithms. For the disparity is not limited to the integer. The real world is continuous, but not segmented. Therefore, the traditional algorithm often uses the sub-pixel technology after the initial disparity map has been obtained. This technology can improve the accuracy and makes the image becomes smooth, and without the ladder in the disparity map. Compared to traditional methods the RBFSM is more realistic and convenient. In addition, the new algorithm does not know the disparity range at the beginning. But the algorithm will give the disparity range for each image, as shown below dis_Venus is 18.43, dis_Cones is 53.54, and dis_Teddy is 60.24. While the true disparity ranges are dis_venus is 19, dis_cones is 59, and

dis_teddy is 59. The error is only about 6%. See RBFSM can accurately give the max disparity, so the algorithm does not need the prior knowledge of the disparity range.

Here we use a common evolution mechanism in [10] as described in (18). The $d_c(x,y)$ is the disparity from each algorithm and the $d_T(x,y)$ is the disparity from the groundtruth, and the (x,y) is the pixel point in the left image (*imagel*). In table 1, we give the accuracy for each algorithm's accuracy in different images.

$$a ccuracy = \frac{1}{W \times H} \sum_{(x,y) \in imagel} (|d_c(x,y) - d_T(x,y)| > 1) \tag{18}$$

| a).left image | b).right image | c).groundtruth | d).High brightness | e).Rotation |
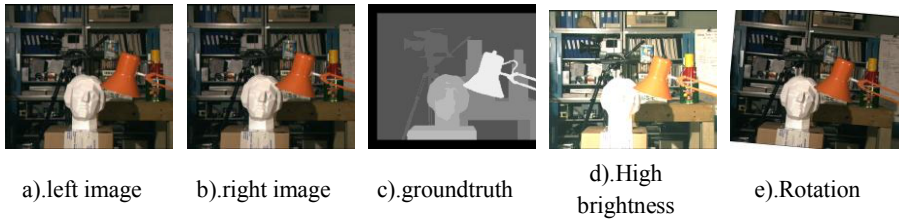
Fig. 3. The Tsukuba in standard and non-standard situation

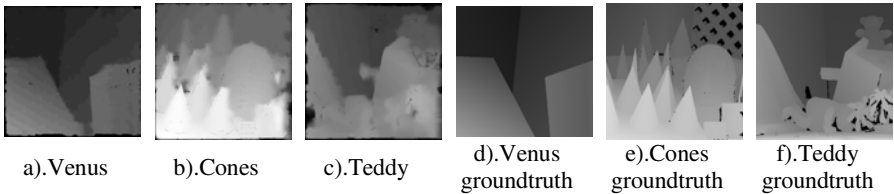| a).Venus | b).Cones | c).Teddy | d).Venus groundtruth | e).Cones groundtruth | f).Teddy groundtruth |

Fig. 4. The RBFSM algorithm for each image and their groundtruth.

Table 1. Each algorithm's accuracy in different images

Image	Algorithm						
	SAD	SO	DP	GC	BP	MSOM [9]	RBFSM
Tsukuba	15.31	7.01	6.42	4.80	6.02	5.17	12.5
Venus	20.02	8.53	13.21	4.11	3.11	3.48	5.06
Cones	38.46	30.05	19.54	22.35	20.20	14.97	13.33
Teddy	34.25	25.36	18.12	20.15	16.85	11.77	10.12

3.3 Non-standard Situation

In this section, we will discuss the Tsukuba in non-standard situations to further reflect the superiority of the algorithm. Considering the two following situations, one

of the Tsukuba images is in high brightness, as the Fig. 3 d) shows. Though the accuracy is decreased now, we still can get a good disparity map in Fig. 5 a). It means that the algorithm is not very sensitive to the intensity. Meanwhile, we try the other algorithms in this case and give the results in Fig. 5 from b) to f), showing that the traditional methods could not get the disparity map. Perhaps it is because the parameter settings, but we can surely know that the traditional methods are not as good as the RBFSM. Another situation is one of the images has a slight rotation, in the experiments we rotate the left image about 5 degrees, as the e) in Fig. 3 shows. This situation is also impossible to deal with using the present algorithm. But the RBFSM can still do it quite well, which is shown in a) in Fig. 6. We also try the traditional methods to this situation from b) to f) in Fig. 6. Just as the above describe, the traditional methods is very poor now. To deal with this situation we should also consider the vertical disparity, so from this experiment we know the traditional methods is weak when there is vertical disparity while the new algorithm can still deal with the rotation. Now the RBFSM algorithm should be expanded in the global search space to give the vertical disparity, and at this time we just need to arrange x from 1 to H in (5). But it will also increase the search space, and make the time consumption increased.
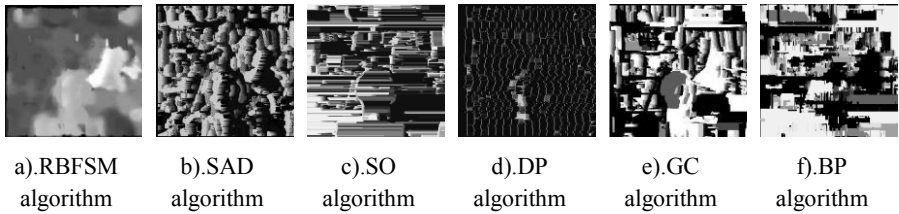

| a).RBFSM algorithm | b).SAD algorithm | c).SO algorithm | d).DP algorithm | e).GC algorithm | f).BP algorithm |

Fig. 5. The disparity map of high brightness images

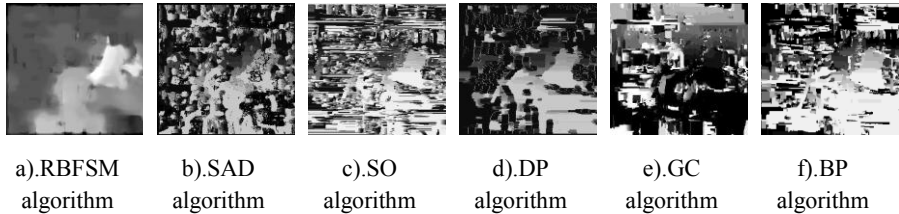| a).RBFSM algorithm | b).SAD algorithm | c).SO algorithm | d).DP algorithm | e).GC algorithm | f).BP algorithm |

Fig. 6. The disparity map of rotation images

4   Conclusions

Because of traditional algorithms have so many shortcomings, here designs a new algorithm called RBFSM. It is based on the RBFNN to deal with the stereo matching. Let the whole image pixels as the input and then their corresponding disparity will be showed in the output. The RBFSM algorithm need not to know the disparity range thus reduces the amount of the redundant search. It uses the constraint between

multiple lines to get a global optimum, so as to avoid the discontinuity between each line. Compared with the algorithm which only considers the best single line, the precision is greatly improved. Without the complexity compute like solving the minimum cut like in GC, thus the time complexity is greatly reduced. The RBFSM has taken various constraints into account, such as intensity constraints, local information constraints, and occlusion constraints, so the robustness is greatly strengthened. When the camera has a vertical displacement, the algorithm still could obtain its corresponding vertical disparity.

Experimental results show the superiority of the RBFSM and prove that it is greatly improved over many traditional methods.

References

1. Richard, S., Ramin, Z.: An Experimental Comparison of Stereo Algorithms. Vision Algorithms: Theory and Practice 1883 (1999)
2. Cassisa, C.: Local VS Global Energy Minimization Methods: Application to Stereo Matching. In: IEEE International Conference on Progress in Informatics and Computing, pp. 678–683 (2010)
3. Ho, Y.J., Kyoung, M.L., Sang, U.L.: Stereo Matching Using Scanline Disparity Discontinuity Optimization. In: Advanced Concepts for Intelligent Vision Systems, pp. 588–597 (2006)
4. Aaron, F.B., Stephen, S.I.: Large Occlusion Stereo. International Journal of Computer Vision 33(3), 181–200 (1999)
5. Yuri, B., Vladimir, K.: An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision. IEEE Transactions on PAMI 26, 1124–1137 (2004)
6. Sun, J., Shum, H.Y., Zheng, N.N.: Stereo Matching Using Belief Propagation. IEEE Transon on PAMI 25(7), 787–800 (2003)
7. Yassine, R.: Multilevel and Neural-network-based Stereo-matching Method for Real-time Obstacle Detection Using Linear Cameras. IEEE Transactions on Intelligent Transportation Systems 6(1), 54–62 (2005)
8. Sun, T.H.: Improving Stereo Matching Quality with Scanline-based Asynchronous Hopfield Neural Network. Journal of the Chinese Institute of Industrial Engineers 24(1), 50–59 (2007)
9. Marco, V., Ignazio, G., Elisabetta, B.: Dense Two-frame Stereo Correspondence by Self-organizing. Neural Network, 1035–1042 (2009)
10. Scharstein, D., Szeliski, R.: A Taxonomy and Evaluation of Dense Two-frame Stereo Correspondence Algorithms. International Journal of Computer Vision 47(1), 7–42 (2002)

An Evolutionary Algorithm Based Optimization of Neural Ensemble Classifiers

Chien-Yuan Chiu and Brijesh Verma

Central Queensland University,
Rockhampton, Qld 4702, Australia
`chien-yuan.chiu@cqumail.com,`
`b.verma@cqu.edu.au`

Abstract. Ensemble classifiers are very useful tools and can be applied in many real world applications for classifying unseen data patterns into one of the known or unknown classes. However, there are many problems facing ensemble classifiers such as finding appropriate number of layers, clusters or even base classifiers which can produce best diversity and accuracy. There has been very little research conducted in this area and there is lack of an automatic method to find these parameters. This paper presents an evolutionary algorithm based approach to identify the optimal number of layers and clusters in hierarchical neural ensemble classifiers. The proposed approach has been evaluated on UCI machine learning benchmark datasets. A comparative analysis of results using the proposed approach and recently published approaches in the literature is presented in this paper.

Keywords: Neural ensemble classifiers, evolutionary algorithms, optimization.

1   Introduction

Ensemble classifiers are approaches which train multiple classifiers and fuse their decisions to produce the final decision. The training process in an ensemble classifier aims to produce the base classifiers in such a way that they are accurate and also differ from each other in terms of the errors they make on identical patterns. This phenomenon is known as diversity [1-4]. A commonly used approach to generate the base classifiers is by training them on different subsets of the data. This ensures diversified learning of the base classifiers and achieves higher accuracy. The subset selection algorithm varies among the different ensemble generation methods. Many ensemble classifier generation methods have been recently developed and there has been lot of research to generate ensemble classifiers using diversity phenomenon. Bagging [5] is one of the oldest ensemble classifier generation techniques. Diversity in conventional bagging is achieved by training the base classifiers on randomly drawn subsets from the training data. Bagging and its variants [6-7] provide a mechanism to achieve diversity but it does not mention any systematic mechanism to identify difficult–to–classify overlapping data. Boosting [8-9] is another popular ensemble classifier generation technique which mainly achieves diversity by creating data subsets using re-sampling of training data. The subsets in boosting, however, not

necessarily contain examples that are difficult to classify when combined together. A number of variants of boosting technique have been recently described in the literature, for example, AdaBoost [9], weighted instance selection method [10], smoothed boosting [11], boosting recombined weak classifiers [12], Learn++.NC [13] and heterogeneous boosting [14].

In additional to convention ensemble classifier generation techniques, some classifier ensemble generation techniques based on multiple clusters idea have been recently published [4, 15-18]. The idea is based on that any data set can contain multiple clusters and some clusters are easy to classify and some are not easy to classify. A clear identification of these difficult to classify clusters can better guide the learning process in ensemble classifiers. This idea has been used in clustered ensembles [4, 15-18] that identify multiple clusters within the classified data and train base classifiers. Although, a data pattern in clustered ensembles can belong to only one cluster and as a result the decision can be obtained from a single classifier leading to lack of diversity. The purpose of the research presented in this paper is to improve the diversity and accuracy through layered clustering and optimising layers and clusters using a novel evolutionary algorithm based approach.

This paper is consists of 4 sections as follows. Section 2 presents the proposed approach including research methodology and processes for training and testing. Section 3 details the benchmark data sets, base classifiers' parameters, experimental results and comparative analysis. The conclusion is presented in Section 4.

2   Proposed Approach

An overview of the proposed research methodology is presented in Fig. 1. The first step is to partition data set into variable number of clusters (1...N) using N layers to achieve diversity. The identical patterns from data set may belong to N alternate clusters. The second step is to train neural network based classifiers on clusters at N layers. The third step is to apply evolutionary algorithm to optimize the parameters.
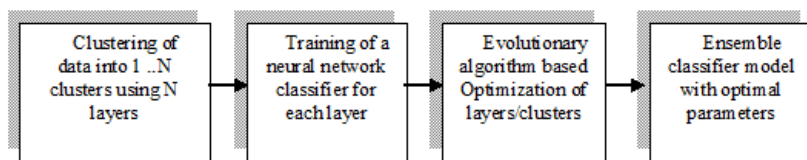

Fig. 1. An overview of the proposed methodology

The main parameters are number of layers (N) and number of clusters (C) at layer l, where l is between 1 and N. The diversity might directly influence classification performance which means that we have to deal with two optimality criteria (accuracy and diversity) while searching for the optimal N and C. An evolutionary algorithm for searching the optimal L and C is used in this paper. The detailed processes used in the proposed ensemble classifier for training/optimising layers and testing are shown below in Fig. 2.

Neural networks as base classifiers are used in this paper but any other classifiers can be used in place of neural networks. The training data set is clustered in N separate layers. At each layer, the data is segmented into clusters based on clustering parameters [16] [20]. The neural network is trained using standard back propagation learning [21]. The parameters settings for the neural networks include variable hidden units between 12 and 60, learning rate of 0.01, momentum term of 0.3 and 0.4 and 25 epochs. The majority voting for decision fusion is used. The diversity was computed using Kohavi-Wolpert (KW) variance [23] [24].
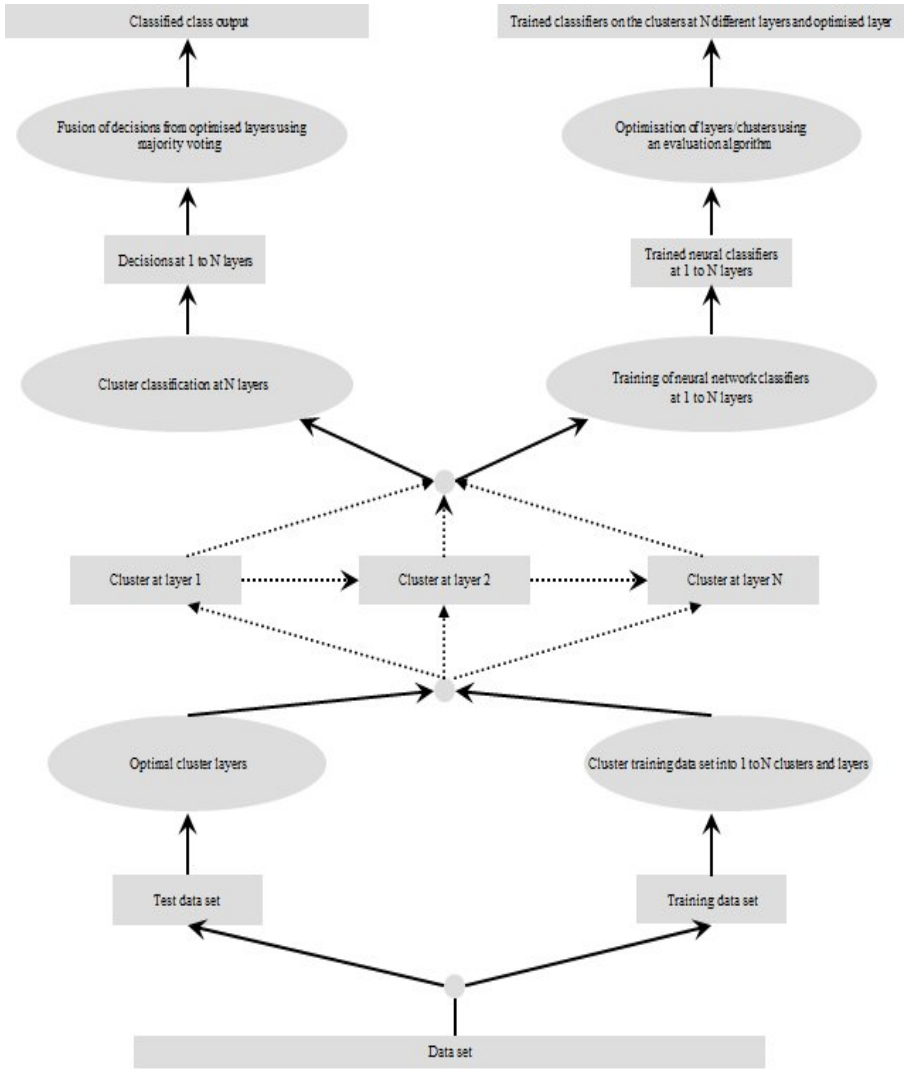
Fig. 2. Architecture of the training and testing processes used in the proposed approach

3   Results and Discussion

The benchmark datasets from UCI Machine Learning Repository [19] has been acquired, processed and used in this research for evaluating the proposed methodology. In all cases, the dataset has been divided into training and test sets. The experiments have been conducted on 8 datasets using these training and test sets. A summary of the datasets used in this paper is presented below in Table 1. In the experiments, 50% of the dataset was used for training and the remaining 50% for the testing. The parameters for training of neural networks as base classifiers are shown in Table 2. The evolutionary algorithm was applied for optimisation of layers and the detailed parameters are given in Table 3. The classification results for bagging and boosting are obtained using WEKA and the same training and test datasets. All the experiments for the proposed methodology were conducted using MATLAB.

Table 1. Data sets used in the experiments

Dataset	Instance	Attribute	Class
Wine	178	13	3
Glass	214	10	7
Ionosphere	351	33	2
Sonar	208	60	2
Thyroid	215	5	3
Iris	150	4	3
Ecoli	336	7	8
Liver	345	6	2

Table 2. The set of parameters for evolutionary algorithm

Population size	Mutation probability	Termination count
60	0.001	12

Table 3. Best parameters for neural networks used in the experiments

Dataset	#Hidden units	Learning rate	Momentum	#Epochs
Wine	28	0.01	0.4	25
Glass	50	0.01	0.5	20
Ionosphere	50	0.01	0.4	25
Sonar	12	0.01	0.4	25
Thyroid	12	0.01	0.4	25
Iris	50	0.01	0.4	25
Ecoli	12	0.01	0.6	25
Liver	12	0.01	0.3	15

The best training and test accuracies obtained by conducting experiments on 8 benchmark datasets using the proposed approach are presented below in Table 4. The table also presents the number of layers, diversity and min/max clusters. Table 5 presents the test accuracies obtained with different neural network training parameters.

Table 4. Best training and test accuracies using the proposed approach

Dataset	Diversity	Training accuracy [%]	Test accuracy [%]	#Of layer	Min clusters	Max clusters
Wine	0.070135	100.00	98.91	88	5	88
Glass	0.072353	97.22	94.55	8	2	97
Ionosphere	0.055437	98.28	96.57	46	2	162
Sonar	0.078193	100.00	87.62	59	3	103
Thyroid	0.018869	100.00	98.18	23	1	110
Iris	0.042774	100.00	97.33	29	1	66
Ecoli	0.078086	92.77	89.41	18	2	166
Liver	0.073340	97.09	68.91	70	5	172

Table 5. Accuracies vs. neural network parameters

Dataset	#Hidden units	Learning rate	Momentum	#Epochs	Test accuracy [%]
Wine	**28**	**0.01**	**0.4**	**25**	**98.89**
	12	0.01	0.3	25	98.89
	30	0.01	0.3	25	98.89
Glass	**50**	**0.01**	**0.5**	**20**	**94.55**
	50	0.01	0.6	25	93.64
	12	0.01	0.5	20	92.27
Ionosphere	**50**	**0.01**	**0.4**	**25**	**96.57**
	50	0.01	0.6	20	95.42
	36	0.001	0.4	10	93.18
Sonar	**50**	**0.01**	**0.4**	**25**	**87.62**
	12	0.01	0.4	25	86.67
	30	0.01	0.5	25	86.67
Thyroid	**12**	**0.01**	**0.4**	**25**	**98.18**
	12	0.01	0.3	25	97.27
	12	0.01	0.3	20	97.27
Iris	**50**	**0.01**	**0.4**	**25**	**97.33**
	12	0.01	0.6	30	97.33
	12	0.01	0.3	25	96.00
Ecoli	**12**	**0.01**	**0.6**	**25**	**89.41**
	12	0.1	0.5	25	88.82
	12	0.01	0.5	20	86.47
Liver	**12**	**0.01**	**0.3**	**15**	**68.91**
	12	0.01	0.3	20	67.06
	12	0.001	0.3	15	65.90

3.1 Comparative Analysis

The Table 6 provides a comparative analysis of results obtained by using the proposed and existing ensemble classifier approaches on UCI machine learning benchmark datasets. The best classification accuracies on the test sets have been used for comparison purposes. The proposed approach using accuracy criteria performance achieved 2.42% better than Bagging and 1.05% better than Boosting. The better performance of the proposed methodology is due to optimised parameters for ensemble classifier.

Table 6. Comparative analysis of the proposed and existing approaches

Dataset	Bagging [%]	Boosting [%]	Proposed approach in this paper [%]
Wine	98.89	100.00	98.89
Glass	90.00	92.73	94.55
Ionosphere	95.45	94.89	96.57
Sonar	80.95	86.67	87.62
Thyroid	94.55	96.36	98.18
Iris	97.33	97.33	97.33
Ecoli	87.65	87.65	89.41
Liver	68.79	68.21	68.91

4   Conclusions

We have presented and investigated a novel approach for optimisation of neural ensemble classifier. The clustering of data and neural trained base classifiers were fused together to find the best ensemble classifier. The proposed approach was implemented and evaluated on 8 machine learning benchmark datasets. The comparative results showed that the proposed approach is 2.42% better than Bagging and 1.05% better than Boosting ensemble classifiers. In our future research, we are planning to conduct experiments on more UCI datasets and include neural network parameters within optimisation process.

References

1. Scherbart, A., Nattkemper, T.W.: The Diversity of Regression Ensembles Combining Bagging and Random Subspace Method. In: International Conference on Neural Information Processing, pp. 911–918 (2008)
2. Kuncheva, L.I., Whitaker, C.J.: Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy. Machine Learning 51, 181–207 (2003)
3. Tang, E.K., Suganthan, P.N., Yao, X.: An Analysis of Diversity Measures. Machine Learning 65, 247–271 (2006)

 4. Rahman, A., Verma, B., Yao, X.: Non-uniform Layered Clustering for Ensemble Classifier Generation and Optimality. In: International Conference on Neural Information Processing, pp. 551–558 (2010)
 5. Breiman, L.: Bagging Predictors. Machine Learning 24, 123–140 (1996)
 6. Chen, L., Kamel, M.S.: A Generalized Adaptive Ensemble Generation and Aggregation Approach for Multiple Classifiers Systems. Pattern Recognition 42, 629–644 (2009)
 7. Nanni, L., Lumini, A.: Fuzzy Bagging: A Novel Ensemble of Classifiers. Pattern Recognition 39, 488–490 (2006)
 8. Schapire, R.E.: The Strength of Weak Learnability. Machine Learning 5, 197–227 (1990)
 9. Freund, Y., Schapire, R.E.: Decision-theoretic Generalization of On-line Learning and an Application to Boosting. Journal of Computer and System Sciences 55, 119–139 (1997)
10. Pedrajas, N.G.: Constructing Ensembles of Classifiers by Means of Weighted Instance Selection. IEEE Transactions on Neural Networks 20, 258–277 (2009)
11. Jin, M., Zhang, J.: Multi-Class Learning by Smoothed Boosting. Machine Learning 67, 207–227 (2007)
12. Rodriguez, J.J., Maudes, J.: Boosting Recombined Weak Classifiers. Pattern Recognition Letters 29, 1049–1059 (2008)
13. Muhlbaier, M.D., Topalis, A., Polikar, R.: Learn++. NC: Combining Ensemble of Classifiers with Dynamically Weighted Consult-and-vote for Efficient Incremental Learning of New Classes. IEEE Transactions on Neural Networks 20, 152–168 (2009)
14. Nascimento, D.S.C., Coelho, A.L.V.: Ensembling Heterogeneous Learning Models with Boosting. In: Int. Conf. on Neural Information Processing, pp. 512–519 (2009)
15. Rokach, L., Maimon, O., Lavi, I.: Space Decomposition in Data Mining: a Clustering Approach. In: Int. Symposium on Methodologies for Intelligent Systems, Maebashi, pp. 24–31 (2003)
16. Kuncheva, L.I.: Switching Between Selection and Fusion in Combining Classifiers: an Experiment. IEEE Transactions on Systems, Man and Cybernetics 32, 146–156 (2002)
17. Nasierding, G., Tsoumakas, G., Kouzani, A.Z.: Clustering based Multi-Label Classification for Image Annotation and Retrieval. IEEE Int. Conference on Systems, Man and Cybernetics, 4514–4519 (2009)
18. Kuncheva, L.I., Rodriguez, J.J., Plumpton, C.O., Linden, D.E., Johnston, S.J.: Random Subspace Ensembles for FMRI Classification. IEEE Transactions on Medical Imaging 29, 531–542 (2010)
19. Frank, A., Asuncion, A.: UCI Machine Learning Repository: University of California, School of Information and Computer Science, http://archive.ics.uci.edu/ml
20. Ko, A.H.R.R., Sabourin, A., de Britto, S., Oliverira, L.: Pairwise Fusion Matrix for Combining Classifiers. Pattern Recognition 40, 2198–2210 (2007)
21. Rumelhart, D., Hinton, G., Williams, R.: Learning Internal Representations by Error Propagation. Parallel Distributed Processing 1, 318–363 (1986)
22. Quinlan, J.R.: Bagging, Boosting, and C4.5. In: National Conference on Artificial Intelligence, pp. 725–730 (1996)
23. Kohavi, R., Wolpert, D.H.: Bias Plus Variance Decomposition for Zero-One Loss Functions. In: International Conference on Machine Learning, pp. 275–283 (1996)
24. Wanas, N.M., Dara, R.A., Kamel, M.S.: Adaptive Fusion and Co-Operative Training for Classifier Ensembles. Pattern Recognition 39, 1781–1794 (2006)

Stability Criterion of Discrete-Time Recurrent Neural Networks with Periodic Delays

Xing Yin*, Weigen Wu, and Qianrong Tan

College of Computer, Pan Zhi Hua University, Panzhihua, P.R. China,
10# Airport Road, Panzhihua, Sichuan, P.R. China, 617000
{xingyin11,tqrmei6}@126.com, wwg@mail.pzhu.edu.cn

Abstract. This paper deals with the problem of stability criterion of discrete-time recurrent neural networks with periodic delays. It is written as a discrete-time multi-switched liner system (DMSLS), applying the parameter and time dependent Lyapunov functions we obtain several new sufficient conditions and sufficient conditions for asymptotically stability of these systems.

Keywords: discrete-time recurrent neural networks, Delay, Stability, switched liner system.

1 Introduction

Many biological and artificial neural networks contain inherent time delays, which may cause oscillation and instability[1,2]. And in a hardware implementation of neural networks using the very large-scale integration (VLSI) technology, the time delay will be inevitable and occur in the signal transmission among the neurons, which will affect the stability of the neural system and may lead to some complex dynamic behaviors such as periodic oscillation, bifurcation, or chaos[1,2,3]. Thus it is very important to study the stability of recurrent neural networks with time-varying delays. When we consider a longterm dynamical behaviors of the system and consider seasonality of the changing environment in many realistic systems, the parameters of the system usually will change with time[4,5]. Recently, the periodic parameters of neural networks are considered in the literature[6-17]. It should be pointed out that the delay function is described as a periodic function in [6,7,9,10,12,14,15]. And many works center on continuous-time recurrent neural network(CRNN) with time-varying delays(CRNNwTVD). When the CRNNwTVD is simulated by computer and is implemented digitally, however, it be discretized into discrete-time recurrent neural network(DRNN) with time-varying delays(DRNNwTVD)[18]. Therefore, this letter studies the stability criteria of DRNN with periodic delays(DRNNwPD).

On other hand, time-varying delay has a upper bound, so DRNNwTVD is regard as a sub-DRNNwTVD at every time[18,19]. Therefore, DRNNwTVD be transformed to a discrete-time switched liner system(DSLS) which consists of

---

* Corresponding author.

a family of sub-DRNNwTVD and a switching law that orchestrates switching between them. And [20] introduce the generalized activation function, which belongs to the convex hull of piecewise linear functions(PLF). Simultaneously, because in practice linear function is easily implemented, PLF may be used to approximate other activation functions. So, in this letter we assume that the activation functions is PLF. Obviously, the DRNN with PLF is regard as a DSLS.

Motivated by the previous discussion, the aims of this letter are to study the stability problem of DRNNwTVD. When DRNNwTVD be written as a discrete-time multi-switched liner system(DMSLS), applying the parameter and time dependent Lyapunov functions we obtain several new sufficient conditions and sufficient conditions for stability of DRNNwTVD.

This paper is organized as follows. In section 2, we give some basic definitions. We analyze the stability of the SDHNN in section 3. We study the stability of the DRNNwTVD in section 4. Some examples are given in section 5. The last section offers the conclusion of this paper.

2   Preliminaries

Consider the following discrete-time recurrent neural networks with time-varying delays:

$$u(k+1) = Au(k) + Bg(u(k)) + Cg(u(k-d(k))) + I \qquad (1)$$

where $u(k) = \{u_1(k), u_2(k), \cdots, u_n(k)\} \in R^n$ is the state vector associated with n neurons; $A = diag\{a_1, a_2, \cdots, a_n\}$ is a diagonal matrix with positive entries; B and C are, respectively, the connection weight matrix and the delayed connection weight matrix; I is input vector; $g(u) = \{g_1(u), g_2(u), \cdots, g_n(u)\}$ is the activation function vector. $d(k)$ is the time delay, which is a nonnegative and periodic function with period ω and satisfies $0 \le d(k) \le d$.

The equilibrium points of DRNNwTVD (1) exist by the fixed point theorem. In the following, let $u^\circ = \{u_1^\circ, u_2^\circ, \cdots, u_n^\circ\}$ be the equilibrium point of (1), then $x(\cdot) = u(\cdot) - u^\circ$. The system (1) is shifted to the following form:

$$x(k+1) = Ax(k) + Bg(x(k)) + Cg(x(k-d(k))) \qquad (2)$$

Throughout this letter, we have the following assumptions.

A1)$d(k + \omega) = d(k)$ with period ω.

A2) The activation functions in (1), $g_i(.)$, is defined as[]:

$$g_i(x_i) = \begin{cases} g_{iN_i}x_i + e_{iN_i}, & \text{if } x_i \in (-\infty, -b_{iN_i}]; \\ \vdots \\ g_{i1}x_i + e_{i1}, & \text{if } x_i \in [-b_{i2}, -b_{i1}]; \\ g_{i0}x_i, & \text{if } x_i \in [-b_{i1}, b_{i1}]; \\ g_{i1}x_i + e_{i1}, & \text{if } x_i \in [b_{i1}, b_{i2}]; \\ \vdots \\ g_{iN_i}x_i + e_{iN_i}, & \text{if } x_i \in [b_{iN_i}, \infty). \end{cases} \qquad (3)$$

Obviously, from A2) we have:

$$\frac{g_i(\theta_1) - g_i(\theta_2)}{\theta_1 - \theta_2} = g_{i0}, \forall \theta_1, \theta_2 \in [-b_{i1}, b_1] \tag{4}$$

$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots$$

$$\frac{g_i(\theta_1) - g_i(\theta_2)}{\theta_1 - \theta_2} = g_{ih}, \forall \theta_1, \theta_2 \in [-b_{i(h+1)}, -b_{ih}] \cup [b_{ih}, b_{i(h+1)}] \tag{5}$$

$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots$$

$$\frac{g_i(\theta_1) - g_i(\theta_2)}{\theta_1 - \theta_2} = g_{iN_i}, \forall \theta_1, \theta_2 \in (-\infty, -b_{iN_i}] \cup [b_{iN_i}, \infty) \tag{6}$$

And there exists $G(k) = diag\{g_1(k), \cdots, g_i(k), \cdots, g_n(k)\}, g_i(k) \in \{g_{i0}, \cdots, g_{iN_i}\}$ such that $g(x(k)) = G(k)x(k)$ and $g(x(k-d(k))) = G(k)x(k-d(k))$. Then we know $G(k) \in \mathbf{G} = \{\{g_{10}, \cdots, g_{1N_1}\} \times \{g_{20}, \cdots, g_{2N_2}\} \times \cdots \times \{g_{n0}, \cdots, g_{nN_n}\}\}$ and $card(\mathbf{G}) = m = \prod_{i=1}^{n} N_i$. So, (3) can be rewritten as a switched system:

$$x(k+1) = Ax(k) + BG_{\varphi_1(k)}x(k) + CG_{\varphi_2(k)}x(k-d(k)) \tag{7}$$

where we denote by M the set of values the switching activation function $\varphi_1(k)$ and $\varphi_2(k)$ may take, $\varphi_1 : Z^+ \longrightarrow M; \varphi_2 : Z^+ \longrightarrow M, M = \{1, 2, \cdots, m\}$.

The next, similar to [19,20], discrete-time systems with time varying delays is transformed into a switched system. Considering the augmented state vector:

$$z(k) = [x^T(k), x^T(k-1), \cdots, x^T(k-d)]^T \tag{8}$$

The dynamics of the system (7) can be represented as the following switched system:

$$z(k+1) = \Lambda_{\varphi(k)\sigma(k)} z(k) \tag{9}$$

The state matrix $\Lambda_{\varphi(k)\sigma(k)}$ switches in the set of possible matrices $\{\Lambda_{11}, \Lambda_{12}, \cdots, \Lambda_{m^2 d}\}$ according to $\varphi(k) = (\varphi_1(k) - 1)m + \varphi_2(k)$ and the parameter $\sigma(k)$ called switching delay function. We denote that switching activation function $\varphi(k)$ may take $\varphi : Z^+ \longrightarrow M^2, M^2 = \{1, 2, \cdots, m^2\}$ and the periodic delay switching function $\sigma(k)$, which satisfied $\sigma(k+\omega) = \sigma(k)$, may take, $\sigma : Z^+ \longrightarrow D, D = \{1, 2, \cdots, d\}$. In the case of time delays systems, the matrices $\Lambda_{lf}, l = (i-1)m+j, i, j \in M, l \in M^2, f \in D$ are given by:

$$\Lambda_{lf} = \begin{bmatrix} A+BG_i & 0 & \cdots & 0 & CG_j & 0 & \cdots & 0 \\ E & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & E & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & E & 0 \end{bmatrix} \tag{10}$$

3   Main Results

According to the periodic delay switching function, periodic switching sequence is $\sigma, \sigma \in \{s \in D^* : |s| = \omega\}$, where the set of all sequences over an alphabet D

is denoted by D^* and the length of a sequence s is denoted by $|s|$. σ is denoted by $\sigma(k+1)\sigma(k+2)\cdots\sigma(k+\omega)$, where the value of $\sigma(k+f)$, where $1 \le f \le \omega$ and $\sigma(k+f) \in D$, is the symbol in the fth position of σ. D' is denoted by $\{1, 2, \cdots, \omega\}$. Suppose that $\sigma = 12\cdots\omega$ and we define the switching indicator function

$$\xi(k) = [\xi_{lp}^1(k)]_{m^2 \times m^2} \bigotimes [\xi_{fh}^2(k)]_{\omega \times \omega} \tag{11}$$

with $\xi_{lp}^1(k) = \begin{cases} 1, & \text{when the activation function } l \text{ switches to } p; \\ 0, & \text{otherwise.} \end{cases}$, and

$\xi_{fh}^2(k) = \begin{cases} 1, & \text{when the periodic delay } f \text{ switches to } h, \ h = (f+1) mod\omega; \\ 0, & \text{otherwise.} \end{cases}$ and

the subsystem indicator function

$$\xi(k) = [\xi_1^1(k), \cdots, \xi_l^1(k), \cdots, \xi_{m^2}^1(k)]^T \bigotimes [\xi_1^2(k), \cdots, \xi_f^2(k), \cdots, \xi_\omega^2(k)] \tag{12}$$

with $\xi_l^1(k) = \sum\limits_{p=1}^{m^2} \xi_{lp}^1(k)$ and $\xi_f^2(k) = \sum\limits_{h=1}^{\omega} \xi_{fh}^2(k)$

Proposition 3.1: The following statements are equivalent:

1) There exists a Lyapunov function:

$$V(k, x(k)) = z^T(k)(\sum\limits_{l=1}^{m^2} \sum\limits_{f=1}^{\omega} \xi_l^1(k)\xi_f^2(k)P_{lf})z(k))$$

whose difference is negative definite and where $\xi_l^1(k)$ and $\xi_f^2(k)$ are defined in (12), proving asymptotic stability of the system (9).

2) There exist ωm^2 symmetric matrices $P_{lf} = P_{lf}^T > 0, \forall l \in M^2, \forall f \in D'$, satisfying the LMIs:

$$\begin{bmatrix} P_{lf} & \Lambda_{lf}P_{ph} \\ * & P_{ph} \end{bmatrix} > 0, \forall(l, p) \in M^2 \times M^2; \forall f \in D', h = (f+1) mod\omega \tag{13}$$

Proof: The switched system (9) can also be written as

$$z(k+1) = \sum\limits_{l=1}^{m^2} \sum\limits_{f=1}^{\omega} \xi_l^1(k)\xi_f^2(k)\Lambda_{lf}z(k) \tag{14}$$

From above, it is easy to verify that the system (9) is asymptotically stable if and only if the system (14) is asymptotically stable. By the results of Theorem 2 in [21], we know that Theorem 3.1 is fulfilled.

Theorem 3.2: The following statements are equivalent:

1) There exists a Lyapunov function:

$$V(t, x(t)) = z^T(t) \sum\limits_{f=1}^{m^{2\omega}} \rho_i(t)P_i)z(t)$$

whose difference is negative definite and where $\rho_i(t)$ is defined in (19), proving asymptotic stability of the system (9).

2) There exist $m^{2\omega}$ symmetric matrices $P_i = P_i^T > 0, \forall i \in M^{2\omega}$, satisfying the LMIs:

$$\begin{bmatrix} P_i & \Psi_i^T P_j \\ * & P_j \end{bmatrix} > 0, \forall (i,j) \in M^{2\omega} \times M^{2\omega} \tag{15}$$

with $\Psi_i = \prod_{f=1}^{\omega} \Lambda_{l_f f}$, which $i = \sum_{f=1}^{\omega} (l_f - 1)m^{2(f-1)}, l_f \in M^2$

Proof: Assume that $t = (\lfloor k/\omega \rfloor - 1) \cdot \omega$ and time step of t is ω that $t+1 = \lfloor k/\omega \rfloor \cdot \omega$. suppose that $u = t+1$, $\overline{u} = t$ and $\overline{\Psi} = \prod_{o=u+1}^{k+1} (\sum_{l=1}^{m^2} \xi_l^1(o) \Lambda_{l\sigma(o)})$ Obviously, the switched system (9) can be written as

$$z(k+1) = \overline{\Psi} z(t+1) = \overline{\Psi} \Psi_{\alpha(t)} z(t) \tag{16}$$

where $\alpha(t)$ is a switching rule defined by $\alpha(t) :: Z^+ \longrightarrow \Omega, \Omega = \{1, 2, \cdots, m^{2\omega}\}$ and $\Psi_{\alpha(t)} = \prod_{o=\overline{u}+1}^{t+1} (\sum_{l=1}^{m^2} \xi_l^1(o) \Lambda_{l\sigma(o)})$

Because $\overline{\Psi}$ is a constant matrix, the switched system (16) can be translated into

$$z(t+1) = \Psi_{\alpha(t)} z(t) \tag{17}$$

Suppose that $\sigma = 12 \cdots \omega$ and we define the switching indicator function:

$$\rho(t) = [\xi_{l_1 l_2}^1(\overline{u}+1)] \bigotimes \cdots \bigotimes [\xi_{l_f l_{f+1}}^1(\overline{u}+f)] \bigotimes \cdots \bigotimes [\xi_{l_\omega l_1}^1(\overline{u}+\omega)] \tag{18}$$

and the subsystem indicator function

$$\begin{aligned} \rho(t) &= [\rho_1(t), \cdots, \rho_i(t), \cdots, \rho_{m^{2\omega}}(t)]^T = [\rho_{ij}(t)]_{m^{2\omega} \times m^{2\omega}} \\ &= [\xi_1^1(\overline{u}+1), \cdots, \xi_{l_1}^1(\overline{u}+1), \cdots, \xi_{m^2}^1(\overline{u}+1)]^T \bigotimes \\ &\quad \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ &\quad [\xi_1^1(\overline{u}+f), \cdots, \xi_{l_f}^1(\overline{u}+f), \cdots, \xi_{m^2}^1(\overline{u}+f)]^T \bigotimes \\ &\quad \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ &\quad [\xi_1^1(\omega), \cdots, \xi_{l_\omega}^1(\omega), \cdots, \xi_{m^2}^1(\omega)]^T \end{aligned} \tag{19}$$

with $\rho_i(t) = \sum_{j=1}^{m^{2\omega}} \rho_{ij}(t)$ and $\xi_{l_f}^1(\overline{u}+f) = \sum_{l_{f+1}=1}^{m^2} \xi_{l_f l_{f+1}}^1(\overline{u}+f)$

Then, the switched system (17) can also be written as

$$z(t+1) = \sum_{i=1}^{m^{2\omega}} \rho_i(t) \Psi_i z(t) \tag{20}$$

From above, it is easy to verify that the system (9) is asymptotically stable if and only if the system (20) is asymptotically stable. By the results of Theorem 2 in [21], we know that Theorem 3.1 is fulfilled.

Remark 1: When $M^2 = \{1\}$, suppose that $G_{\varphi_1(k)} = G_{\varphi_2(k)} = G$, the system (10) is written as:

$$z(k+1) = \Lambda_{\sigma(k)} z(k) \qquad (21)$$

where $\sigma(k)$ is defined in (9). Based on the proof of Proposition 3.1, we suppose that periodic switching sequence $\sigma = \sigma(k+1)\sigma(k+2)\cdots\sigma(k+\omega) = 12\cdots\omega$. Then we have the following result:

Corollary: The following statements are equivalent:

1) There exists a Lyapunov function: $V(t, x(t)) = z^T(t)Pz(t))$ whose difference is negative definite, proving asymptotic stability of the system (21).

2) There exist a symmetric matrices $P = P^T > 0$, satisfying the LMI: $\begin{bmatrix} P & \Psi^T P \\ * & P \end{bmatrix} > 0$, with $\Psi = \prod_{f=1}^{\omega} \Lambda_{1f}$.

4   Conclusions

This paper was dedicated to the delay-dependent stability of uncertain discrete-time recurrent neural networks with time-varying delay. A less conservative LMI-based globally stability criterion is obtained with switched quadratic Lyapunov functional approach and free-weighting matrix approach for periodic uncertain discrete-time recurrent neural networks with a time-varying delay.

References

1. Cao, J.D., Yuan, K., Li, H.X.: Global asymptotical stability of recurrent neural networks with multiple discrete delays and distributed delays. IEEE Trans. Neural Netw. 17(6), 1646–1651 (2006)
2. Li, T., Guo, L., Sun, C., Lin, C.: Further Results on Delay-Dependent Stability Criteria of Neural Networks With Time-Varying Delays. IEEE Trans. on Neural Networks 19(4), 726–730 (2008)
3. Huang, H., Feng, G., Cao, J.D.: Robust state estimation for uncertain neural networks with time-varying delay. IEEE Trans. on Neural networks 19(8), 1329–1339 (2008)
4. Xiang, H., Yan, K.M., Wang, B.Y.: Existence and global exponential stability of periodic solution for delayed discrete high-order Hopfield-type neural networks. Physics Letters A 352, 341–349 (2006)
5. Xiang, H., Yan, K.M., Wang, B.Y.: Existence and global exponential stability of periodic solution for delayed high-order Hopfield-type neural networks. Discrete Dynamics in Nature and Society 2005(3), 281–297 (2005)

6. Huang, C.X., He, Y.G., Huang, L.H., Lai, M.Y.: Global exponential periodicity of three-unit neural networks in a ring with time-varying delays. Neurocomputing 71, 1595–1603 (2008)
7. Jiang, H., Cao, J.D.: Global exponential stability of periodic neural networks with time-varying delays. Neurocomputing 70, 343–350 (2006)
8. Shao, Y., Dai, B.: The existence of exponential periodic attractor of impulsive BAM neural network with periodic coefficients and distributed delays. Neurocomputing 73, 3123–3131 (2010)
9. Liu, Y., You, Z., Cao, L.: On the almost periodic solution of generalized Hopfield neural networks with time-varying delays. Neurocomputing 69, 1760–1767 (2006)
10. Liu, B., Huang, L.: Existence and exponential stability of almost periodic solutions for Hopfield neural networks with delays. Neurocomputing 68, 196–207 (2005)
11. Cao, J.D., Wang, J.: Global Exponential Stability and Periodicity of Recurrent Neural Networks With Time Delays. IEEE Trans. on circuits and syst. I 52, 920–931 (2005)
12. Lou, X., Cui, B.: Delay-Dependent Criteria for Global Robust Periodicity of Uncertain Switched Recurrent Neural Networks With Time-Varying Delay. IEEE Trans. on Neural Networks 19, 549–557 (2008)
13. Huang, Z., Wang, X., Feng, C.: Multiperiodicity of Periodically Oscillated Discrete-Time Neural Networks with Transient Excitatory Self-Connections and Sigmoidal Nonlinearities. IEEE Trans. on Neural Networks 21, 1643–1655 (2010)
14. Allegretto, W., Papini, D., Forti, M.: Common Asymptotic Behavior of Solutions and Almost Periodicity for Discontinuous, Delayed, and Impulsive Neural Networks. IEEE Trans. on Neural Networks 21, 1110–1125 (2010)
15. Zhang, L., Zhang, Y., Yu, J.: Multiperiodicity and Attractivity of Delayed Recurrent Neural Networks With Unsaturating Piecewise Linear Transfer Functions. IEEE Trans. on Neural Networks 19, 158–167 (2008)
16. Chen, B., Wang, J.: Global exponential periodicity of a class of recurrent neural networks with oscillating parameters and time-varying delays. IEEE Trans. on Neural Networks 16(6), 1440–1449 (2005)
17. Wu, X.R., Wang, Y.N., Huang, L.H., Zuo, Y.: Robust exponential stability criterion for uncertain neural network swith discontinuous activation functions and time-varying delay. Neurocomputing 73, 1265–1271 (2010)
18. Song, Q., Cao, J.D.: Global Dissipativity on Uncertain Discrete-Time Neural Networks with Time-Varying Delays. Discrete Dynamics in Nature and Society (2010), Article ID 810408, 19 pages (2010)
19. Arzen, K.E., Bicchi, A., Hailes, S., Johansson, K.H., Lygeros, J.: On the design and control of wireless networked embedded systems. In: Proceedings of the 2006 IEEE Computer Aided Control Systems Design Symposium, Munich, Germany (2006)
20. Hetel, L., Daafouz, J., Iung, C.: Equivalence between the LyapunovCKrasovskii functionals approach for discrete delay systems and that of the stability conditions for switched systems. Nonlinear Analysis: Hybrid Systems 2, 697–705 (2008)
21. Xu, J., Cao, Y., Sun, Y., Tang, J.: Absolute Exponential Stability of Recurrent Neural Networks With Generalized Activation Function. IEEE Trans. on Neural Networks 19(6), 1075–1089 (2008)
22. Daafouz, J., Riedinger, P., Iung, C.: Stability analysis and control synthesis for switched systems: a switched Lyapunov function approach. IEEE Trans. on Automatic Control 47(11), 1883–1887 (2002)

23. Li, J., Diao, Y.F., Li, M.D., Yin, X.: Stability analysis of Discrete Hopfield Neural Networks with the nonnegative definite monotone increasing weight function matrix. Discrete Dynamics in Nature and Society 2009,Article ID 673548, 10 (2009)
24. Li, J., Diao, Y., Mao, J., Zhang, Y., Yin, X.: Stability Analysis of Discrete Hopfield Neural Networks with Weight Function Matrix. In: Kang, L., Cai, Z., Yan, X., Liu, Y. (eds.) ISICA 2008. LNCS, vol. 5370, pp. 760–768. Springer, Heidelberg (2008)
25. Li, J., Wu, W.G., Yuan, J.M., Tan, Q.R., Yin, X.: Delay-dependent stability criterion of arbitrary switched linear systems with time-varying delay. Discrete Dynamics in Nature and Society 2010, Article ID 347129, 16 (2010)
26. Li, J., Yang, J., Wu, W.G.: Stability analysis of discrete Hopfield neural networks with column arbitrary-dominant weight matrix. Neurocomputing (revised manuscript submitted to, 2011)

Improved Global Robust Stability Criteria for Delayed BAM Neural Networks

Xiaolin Li and Ming Liu

Department of Mathematics, Shanghai University, Shanghai 200444, China
{xlli,herry_shanghai}@shu.edu.cn

Abstract. This paper is concerned with uniqueness and global robust stability for the equilibrium point of the interval bidirectional associative memory (BAM) delayed neural networks. By employing linear matrix inequality and Lyapunov functional, a new criterion is proposed for the global robust stability of BAM neural networks. An example is given to show the effectiveness of the present results.

1 Introduction

Kosko proposed a new class of neural networks called bidirectional associative memory (BAM) neural networks[1-3]. The BAM neural network is composed of neurons arranged in two layers, and the neurons in one layer are fully interconnected to the neurons in the other layer. Because of these models are all about the single-layer auto-associative circuit, recent years, the convergence of dynamics of BAM neural network have been extensively studied due to the good application prospects in the areas of pattern recognition, signal, automatic control engineering and image etc.

In that applications stability and the convergence dynamics of BAM neural networks are prerequisite. therefore, a lots of results for BAM neural networks concerning the existence of equilibrium points, global stability have been done, we refer to [4-10,14-16].

A lot of effictive criterions are derived for the global stability in delayed BAM networks by constructing suitable Lyapunov functional and the inequality technique. Cao and Wang [7] study the periodic oscillatory solution of BAM networks with delays. In[8], the author derived several criteria for the global exponential stability of delayed BAM neural networks. And Liao and Wong [9] gave some sufficient conditions for the global robust asymptotic stability of the continuous-time BAM nerual networks. In this paper, we give new criteria to guarantee the global robust stability of the BAM neural networks.

We consider the delayed BAM networks described by the following functional differential equations:

$$
\begin{cases}
\dot{u}_i(t) = -a_i u_i(t) + \sum_{j=1}^{m} w_{ij} f_j(v_j(t - \tau_{ij})) + I_i, \\
\dot{v}_j(t) = -b_j v_j(t) + \sum_{i=1}^{n} v_{ji} g_i(u_i(t - \sigma_{ji})) + J_j,
\end{cases}
\tag{1}
$$

where $i = 1, 2, ..., n$; $j = 1, 2, ..., p$; $u_i(t)$, $v_j(t)$ denote the potential (or voltage) of the cell i and j at time t respectively; a_i, b_j are positive constants, they denote the rate with which the cell i and j reset their potential to the resting state when isolated from the other cells and inputs; time delays τ_{ij}, σ_{ji} are non-negative constants, they correspond to finite speed of axonal signal transmission; the connection weights w_{ij}, v_{ji} are real numbers, they denote the strengths of connectivity between the cells i and j at time $t - \tau_{ij}$ and $t - \sigma_{ji}$ respectively; I_i, J_j denote the ith and jth component of an external input source introduced from outside the network to cell i and j respectively; we denote $u(t) = [u_1(t), u_2(t), ..., u_n(t)] \in R^n$, $v(t) = [v_1(t), v_2(t), ..., v_m(t)] \in R^m$; $\tau = \max\limits_{1 \leq i \leq n, 1 \leq j \leq m} (\tau_{ij})$, $\sigma = \max\limits_{1 \leq i \leq n, 1 \leq j \leq m} (\sigma_{ji})$, $\tau^* = \max(\tau, \sigma)$.

In this paper, the activation functions are assumed to satisfy the following:

$$(A1) |f_i(\zeta)| \leq M_i, |g_i(\zeta)| \leq N_j, \forall \zeta \in R; M_i, N_j > 0.$$

$$(A2) \frac{|f_j(x) - f_j(y)|}{|x - y|} \leq L_i, \frac{|g_j(x) - g_j(y)|}{|x - y|} \leq \widetilde{L_j}.$$

Where $i = 1, 2, ..., n; j = 1, 2, ..., m;$ $u, v \in R, u \neq v$ and $L_i, \widetilde{L_j}$ are positive constants.

The quantities a_i, b_j, w_{ij}, v_{ji} may be intervalized as follows:

$$\begin{aligned}
A_I &= \{A = diag(a_i) : \underline{A} \leq A \leq \overline{A}, i.e., \\
&\underline{a_i} \leq a_i \leq \overline{a_i}, i = 1, 2, \cdots, n, \forall A \in A_I\} \\
B_I &= \{B = (b_j)_{n \times n} : \underline{B} \leq B \leq \overline{B}, i.e., \\
&\underline{b_j} \leq b_{ij} \leq \overline{b_j}, j = 1, 2, \cdots, n, \forall B \in B_I\} \\
W_I &= \{W = (w_{ij})_{n \times n} : \underline{W} \leq W \leq \overline{W}, i.e., \\
&\underline{w_{ij}} \leq w_{ij} \leq \overline{w_{ij}}, i, j = 1, 2, \cdots, n, \forall W \in W_I\}. \\
V_I &= \{V = (v_{ji})_{n \times n} : \underline{V} \leq V \leq \overline{V}, i.e., \\
&\underline{v_{ji}} \leq v_{ji} \leq \overline{v_{ji}}, i, j = 1, 2, \cdots, n, \forall V \in V_I\}.
\end{aligned} \tag{2}$$

2   Preliminaries

The system (1) can be rewritten in the vector form as

$$\begin{cases} \dot{u}(t) = -Au(t) + Wg(v(t - \tau)) + I, \\ \dot{v}(t) = -Bv(t) + Vf(u(t - \sigma)) + J. \end{cases} \tag{3}$$

where $u(t) = (u_1(t), u_2(t), ..., u_n(t))^T$, $v(t) = (v_1(t), v_2(t), ...v_m(t))^T$,
$A = diag(a_1, a_2, ..., a_n)$, $B = diag(b_1, b_2, ..., b_m)$, $W = (w_{ij})_{m \times n}$, $V = (v_{ij})_{n \times m}$,
$f = (f_1, f_2, ..., f_m)^T$, $g = (g_1, g_2, ..., g_n)^T$, $I = (I_1, I_2, ..., I_n)$, $J = (J_1, J_2, ..., J_m)$.

It is clear that under the assumption $(A1)$ and $(A2)$, system (1) has at least one equilibrium. In order to simplify our proof ,we will shift the equilibrium point $u^* = (u_1, u_2, ..., u_n)^T, v^* = (v_1, v_2, ..., v_m)^T$ of system (3) to the origin by

letting $x(t) = u(t) - u^*, y(t) = v(t) - v^*$, and then, the transformed system is as follows:

$$\begin{cases} \dot{x}(t) = -Ax(t) + Wg(y(t-\tau)), \\ \dot{y}(t) = -By(t) + Vf(x(t-\sigma)). \end{cases} \tag{4}$$

From assumption $(A1)$ and $(A2)$ we have the vector $f(x), g(y)$ possesses the following properties:

$$\frac{f(x)}{x} \le L; \frac{g(y)}{y} \le \widetilde{L} \tag{5}$$

Definition 1 [12]: The system given by (1) with the parameter ranges defined by (3) is globally robust stable if there is the unique equilibrium point $(x^*) = (x_1^*, x_2^*, \cdots, x_n^*)^T; (y^*) = (y_1^*, y_2^*, \cdots, y_n^*)^T$ of the system, which is globally asymptotically stable for all $A \in A_I, B \in B_I, W \in W_I, V \in V_I$.

Motivated by the preceding discussion, it suffices to prove the global robust stability of the origin of system (4) with the parameter ranges defined by (2). On the other hand, for real symmetric matrix X and Y, the notation $X \ge Y$ (respectively, $X > Y$) means that the matrix $X - Y$ is positive definite(respectively, semi-define). For $x = [x_1, x_2, \cdots, x_n]^T \in R^n$, let $|x|$ denote the absolute-value vector given by $|x| = [|x_1|, |x_2|, \cdots, |x_n|]^T$. For a matrix $A = (a_{ij})_{(n \times n)} \in R^{n \times n}$, let $|A|$ denote the absolute-value matrix given by $|A| = (|a_{ij}|)_{n \times n}$, $L = diag(L_1, L_2, \cdots, L_n)$.

Lemma 1 [17]: For any $x = [x_1, x_2, \cdots, x_n]^T$, $y = [y_1, y_2, \cdots, y_n]^T$, $B = (b_{ij})_{n \times n}$, $C = (c_{ij})_{n \times n}$ with $|b_{ij}| \le c_{ij}$, we have

$$x^T By \le |x|^T C|y|. \tag{6}$$

3   Main Result

Theorem 1: Under the assumptions $(A1)$ and $(A2)$, system (4) is globally robust stable if there exits six positive diagonal matrices $H_1, H_2, S_1, S_2, P_1, P_2$, and the two positive definite matrices P_2, Q_2 such that the following LMIs hold:

$$\begin{bmatrix} \varPi_1 & 0 & 0 & 0 & 0 & P_1W_0 \\ 0 & \varPi_2 & 0 & 0 & Q_1V_0 & 0 \\ 0 & 0 & -Q_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -P_2 & 0 & 0 \\ 0 & V_0^T Q_1 & 0 & 0 & -Q_2 + S_1 & 0 \\ W_0^T P_1 & 0 & 0 & 0 & 0 & -P_2 + S_2 \end{bmatrix} < 0 \tag{7}$$

and

$$\begin{bmatrix} -H_1 & 0 & 0 & 0 & 0 & P_1U_1 \\ 0 & -H_2 & 0 & 0 & Q_1U_2 & 0 \\ 0 & 0 & -Q_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -P_2 & 0 & 0 \\ 0 & U_2^T Q_1 & 0 & 0 & -S_1 & 0 \\ U_2^T P_1 & 0 & 0 & 0 & 0 & -S_2 \end{bmatrix} < 0 \tag{8}$$

where

$$\Pi_1 = -2P_1\underline{A} + H_1 + 3\widetilde{L}^2 Q_2,$$

$$\Pi_2 = -2Q_1\underline{B} + H_2 + 3L^2 P_2,$$

$W_0 = \frac{1}{2}(\overline{W} + \underline{W}),\ U_1 = \frac{1}{2}(\overline{W} - \underline{W}),\ V_0 = \frac{1}{2}(\overline{V} + \underline{V}),\ U_2 = \frac{1}{2}(\overline{V} - \underline{V}).$

Proof. Consider the following Lyapunov functional:

$$V(x,y) = x^T(t)P_1 x(t) + y^T(t)P_1 y(t) \\ + \int_{t-\tau}^{t} g^T(y(\zeta))P_2 g(y(\zeta))d\zeta + \int_{t-\sigma}^{t} f^T(x(\eta))Q_2 f(x(\eta))d\eta. \tag{9}$$

The time derivative of along the trajectories of (4) takes the form:

$$\dot{V}(x,y) = -2x^T(t)P_1 A x(t) + 2x^T P_1 W g(y(t-\tau)) \\ -2y^T(t)Q_1 B y(t) + 2y^T Q_1 V f(x(t-\tau)) \\ +g^T(y(t))P_2 g(y(t)) - g^T y((t-\tau))P_2 g(y(t-\tau)) \\ +f^T(x(t))Q_2 f(x(t)) - f^T x((t-\sigma))Q_2 f(x(T-\sigma))$$

Since A, P_1, B, Q_1 are positive diagonal matrices, we can get that:

$$-2x^T(t)P_1 A x(t) \le -2x^T(t)P_1\underline{A}x(t). \tag{10}$$

$$-2y^T(t)Q_1 B y(t) \le -2y^T(t)Q_1\underline{B}y(t). \tag{11}$$

Also, we consist $\Delta W = W - W_0$; $\Delta V = V - V_0$. From (2) we can obtain $|\Delta w_{ij}| \le u_{ij}^1$; $|\Delta v_{ij}| \le u_{ji}^2$.

So, we have

$$|P_1 \Delta W| \le PU_1; |Q_1 \Delta V| \le PU_2$$

By lemma1, we have

$$x^T(t)P_1 W g(y(t-\tau)) = x^T(t)P_1 W_0 g(y(t-\tau)) + x^T(t)P_1 \Delta W g(y(t-\tau)) \\ \le x^T(t)P_1 W_0 g(y(t-\tau)) + |x(t)|^T P_1 U_1 |g(y(t-\tau))|. \tag{12}$$

Similarly,we can obtain that

$$y^T(t)Q_1 V f(x(t-\tau)) = y^T(t)Q_1 V_0 f(x(t-\sigma)) + y^T(t)Q_1 \Delta V f(x(t-\sigma)) \\ \le y^T(t)Q_1 V_0 f(x(t-\sigma)) + |y(t)|^T Q_1 U_2 |f(x(t-\sigma))|. \tag{13}$$

Obviously, it is easy to see that

$$x^T(t)H_1 x(t) \le \sum_{i=1}^{n} |x_i(t)||h_i^1||x_i(t)| = |x(t)|^T H_1 |x(t)|; \tag{14}$$

$$y^T(t)H_2 y(t) \le \sum_{j=1}^{m} |y_j(t)||h_j^2||y_j(t)| = |y(t)|^T H_2 |y(t)|. \tag{15}$$

And

$$
\begin{aligned}
f^T(x(t-\sigma))S_1 f(x(t-\sigma)) &\leq \sum_{i=1}^{n} |f(x_i(t-\sigma))|s_i^1|f(x_i(t-\sigma))| \\
&= |f(x(t-\sigma))|^T S_1 |f(x(t-\sigma))| \\
g^T(y(t-\tau))S_2 g(y(t-\tau)) &\leq \sum_{i=1}^{n} |g(y_i(t-\tau))|s_i^2|g(y_i(t-\tau))| \\
&= |g(y(t-\tau))|^T S_2 |g(y(t-\tau))|
\end{aligned}
\tag{16}
$$

It follows from (5) that

$$
\begin{aligned}
f^T(x(t))Q_2 f(x(t)) &= -f^T(x(t))Q_2 f(x(t)) - f^T(x(t))|Q_2|f(x(t)) \\
&\quad +3f^T(x(t))Q_2 f(x(t)) \\
&\leq -f^T(x(t))Q_2 f(x(t)) - f^T(x(t))|Q_2|f(x(t)) \\
&\quad +3x^T(t)\widetilde{L}^2 Q_2 x(t); \\
g^T(y(t))P_2 g(y(t)) &= -g^T(y(t))P_2 g(y(t)) - g^T(y(t))|P_2|g(y(t)) \\
&\quad +3g^T(y(t))P_2 g(y(t)) \\
&\leq -g^T(y(t))P_2 g(y(t)) - g^T(y(t))|P_2|g(y(t)) \\
&\quad +3y^T(t)L^2 P_2 y(t)
\end{aligned}
\tag{17}
$$

By using (12)-(17) ,we can get :

$$
\begin{aligned}
\dot{v}(x,y) &\leq x^T(t)(-2P_1\underline{A} + H_1 + 3\widetilde{L}^2 Q_2)x(t) + 2x^T(t)(P_1 W_0)g(y(t-\tau)) \\
&\quad +y^T(t)(-2Q_1\underline{B} + H_2 + 3L^2 P_2)y(t) + 2y^T(t)(Q_1 V_0)f(x(t-\sigma)) \\
&\quad +f^T(x(t))(-Q_2)f(x(t)) + g^T(y(t))(-P_2)g(y(t)) \\
&\quad +f^T(x(t-\sigma))(-Q_2 + S_1)f(x(t-\sigma)) \\
&\quad +g^T(y(t-\tau))(-P_2 + S_2)g(y(t-\tau)) \\
&\quad +|x(t)|^T(-H_1)|x(t)| + 2|x(t)|^T(P_1 U_1)|g(y(t-\tau))| \\
&\quad +|y(t)|^T(-H_2)|y(t)| + 2|y(t)|^T(Q_1 U_2)|f(x(t-\sigma))| \\
&\quad +|f(x(t))|^T(-Q_2)|f(x(t))| + |g(y(t))|^T(-P_2)|g(y(t))| \\
&\quad +|f(x(t-\sigma))|^T(-S_1)|f(x(t-\sigma))| \\
&\quad +|g(y(t-\tau))|^T(-S_2)|g(y(t-\tau))|
\end{aligned}
\tag{18}
$$

Then, we can arrange (18) as

$$
\dot{V}(x(t)) \leq \eta_1^T \Phi_1 \eta_1 + \eta_2^T \Phi_2 \eta_2
\tag{19}
$$

where $\eta_1 = [x^T(t), y^T(t), f^T(x(t)), g^T(y(t)), f^T(x(t-\sigma)), g^T(y(t-\tau))]^T$,
$\eta_2 = |\eta_1|$,and

$$
\Phi_1 = \begin{bmatrix}
-2P_1\underline{A} + H_1 + 3\widetilde{L}^2 Q_2 & 0 & 0 & 0 & 0 & P_1 W_0 \\
0 & -2Q_1\underline{B} + H_2 + 3L^2 P_2 & 0 & 0 & Q_1 V_0 & 0 \\
0 & 0 & -Q_2 & 0 & 0 & 0 \\
0 & 0 & 0 & -P_2 & 0 & 0 \\
0 & V_0^T Q_1 & 0 & 0 & -Q_2 + S_1 & 0 \\
W_0^T P_1 & 0 & 0 & 0 & 0 & -P_2 + S_2
\end{bmatrix}
$$

$$\Phi_2 = \begin{bmatrix} -H_1 & 0 & 0 & 0 & 0 & P_1U_1 \\ 0 & -H_2 & 0 & 0 & Q_1U_2 & 0 \\ 0 & 0 & -Q_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -P_2 & 0 & 0 \\ 0 & U_2^TQ_1 & 0 & 0 & -S_1 & 0 \\ U_2^TP_1 & 0 & 0 & 0 & 0 & -S_2 \end{bmatrix}$$

From the inequality (7) and (8) ,it follows that $\dot{V}(x(t)) < 0$ for any $x(t) \in R^n$, $y(t) \in R^m$ with $x^T(t)x(t) \neq 0$ and $y^T(t)y(t) \neq 0$. Thus, the origin (4) , the equilibrium point u^* and v^* of (1) is globally robust stable.

Now we are in a position to prove the uniqueness of the equilibrium. The equilibrium point (x_1, y_1) of (4) is given by

$$\begin{cases} -Ax_1 + Wg(y_1) = 0, \\ -By_1 + Vf(x_1) = 0. \end{cases} \tag{20}$$

Assume the there exist the equilibrium point satisfying (20). we can define the functional

$$\begin{aligned} F(x_1, y_1) &= 2x_1^T P_1(-Ax_1 + Wg(y_1)) + 2y_1Q_1(-By_1 + Vf(x_1)) \\ &+ x_1^T H_1 x_1 - x_1^T H_1 x_1 + y_1^T H_2 y_1 - y_1^T H_2 y_1 \\ &+ f^T(x)S_1f(x) - f^T(x)S_1f(x) \\ &+ g^T(y)S_2g(y) - g^T(y)S_2g(y) \\ &+ f^T(x)Q_2f(x) - f^T(x)Q_2f(x) \\ &+ g^T(y)P_2g(y) - g^T(y)P_2g(y). \end{aligned} \tag{21}$$

Obviously,$F(x_1, y_1) = 0$.

On the other hand, we can get that

$$\begin{aligned} -2x_1^T P_1 A x_1 &\leq -2x_1^T P_1 \underline{A} x_1, \\ -2y_1^T Q_1 B y_1 &\leq -2y_1^T Q_1 \underline{B} y_1, \\ 2x_1^T P_1 W g(y_1) &\leq 2x_1^T P_1 W_0 g(y_1) + 2|x_1|^T P_1 U_1 |g(y_1)|, \\ 2y_1^T Q_1 V f(x_1) &\leq 2y_1^T Q_1 V_0 f(x_1) + 2|y_1|^T Q_1 U_2 |f(x_1)|, \\ x_1^T H_1 x_1 &= |x_1|^T H_1 |x_1|, \\ y_1^T H_2 y_1 &= |y_1|^T H_2 |y_1|, \\ f^T(x_1)S_1 f(x_1) &= |f(x_1)|^T S_1 |f(x_1)|, \\ g^T(y_1)S_2 g(y_1) &= |g(y_1)|^T S_2 |g(y_1)|. \end{aligned} \tag{22}$$

and

$$\begin{aligned} f^T(x_1)Q_2f(x_1) &\leq -f^T(x_1)Q_2f(x_1) - f^T(x_1)|Q_2|f(x_1) \\ &+ 3x_1^T \widetilde{L}^2 Q_2 x_1, \\ g^T(y_1)P_2g(y_1) &\leq -g^T(y_1)P_2g(y_1) - g^T(y_1)|P_2|g(y_1) \\ &+ 3y_1^T L^2 P_2 y_1. \end{aligned} \tag{23}$$

From (21) and (23) ,we can get that

$$F(x_1, y_1) \leq \widetilde{\eta_1^T} \Phi_1 \widetilde{\eta_1} + \widetilde{\eta_2^T} \Phi_2 \widetilde{\eta_2} \tag{24}$$

where Φ_1 and Φ_2 are defined as (19), and $\widetilde{\eta_1^T} = [x_1^T, y_1^T, f^T(x_1),$ $g^T(y_1), f^T(x_1), g^T(y_1)]^T$, $\eta_2 = |\eta_1|$. Clearly, from (7) and (8), we have $F(x_1, y_1) < 0$. It contradicts with $F(x_1, y_1) = 0$, which means system (4) has the uniqueness of the equilibrium point.

This proof is complete.

4   Example

In the following, we will give an example to consider a delayed BAM neural networks:

$$\underline{A} = \begin{bmatrix} 2\ 0 \\ 0\ 1 \end{bmatrix}, \overline{A} = \begin{bmatrix} 2.05 & 0 \\ 0 & 1.02 \end{bmatrix}, \underline{B} = \begin{bmatrix} 1\ 0 \\ 0\ 1 \end{bmatrix}, \overline{B} = \begin{bmatrix} 1.02 & 0 \\ 0 & 1.03 \end{bmatrix},$$

$$\underline{W} = \begin{bmatrix} 0.020\ 0.041 \\ 0.032\ 0.051 \end{bmatrix}, \overline{W} = \begin{bmatrix} 0.022\ 0.043 \\ 0.034\ 0.053 \end{bmatrix}, \underline{V} = \begin{bmatrix} 0.058\ 0.013 \\ 0.020\ 0.066 \end{bmatrix}, \overline{V} = \begin{bmatrix} 0.060\ 0.015 \\ 0.021\ 0.068 \end{bmatrix},$$

$$I = J = [1 \quad 1]^T, \tau(t) = 0.4 + 0.5\sin(t) \quad \sigma(t) = 0.4 + 0.5\cos(t)$$

and with

$$f(u(t)) = \begin{bmatrix} \tanh x_1(t - \sigma) \\ \tanh 2x_2(t - \sigma) \end{bmatrix}; g(v(t)) = \begin{bmatrix} \tan 5y_1(t - \tau) \\ \tan y_2(t - \tau) \end{bmatrix}.$$

By using the Matlab LMI Control toolbox, it can be easily verified that the LMIs (7) and (8) is feasible and

$$H_1 = \begin{bmatrix} 13.1655 & 0 \\ 0 & 10.6254 \end{bmatrix}, H_2 = \begin{bmatrix} 10.6262 & 0 \\ 0 & 10.6182 \end{bmatrix}, S_1 = \begin{bmatrix} 8.2878 & 0 \\ 0 & 8.2834 \end{bmatrix},$$

$$S_2 = \begin{bmatrix} 9.6728 & 0 \\ 0 & 8.2873 \end{bmatrix}, P_1 = \begin{bmatrix} 17.9339 & 0 \\ 0 & 25.8229 \end{bmatrix}, P_2 = \begin{bmatrix} 15.3144 & 0 \\ 0 & 11.1579 \end{bmatrix},$$

$$Q_1 = \begin{bmatrix} 25.8259 & 0 \\ 0 & 25.7944 \end{bmatrix}, Q_2 = \begin{bmatrix} 11.1591 & 0 \\ 0 & 11.1461 \end{bmatrix}.$$

Therefore, it follows from Theorem 1 the BAM (3) with given parameters is globally robustly stable.

References

1. Kosko, B.: Neurals and Fuzzy System-A Dynamical System Approach to Machine Intelligence. Prentice-Hall, Englewood Cliffs (1992)
2. Kosko, B.: Adaptive bi-directional associative memories. Appl. Opt. 26(23), 4947–4960 (1987)
3. Kosko, B.: Bi-directional associative memories. IEEE Trans. Syst.Man Cybernet. 18(1), 49–60 (1988)

 4. Park, J.H.: Robust stability of bidirectional associative memory neural networks with time delays. Physics Letters A 349, 494–499 (2006)
 5. Lou, X., Cui, B.: Global asymptotic stability of delay BAM neural networks with impulses. Chaos, Solitons and Fractals 29, 1023–1031 (2006)
 6. Cao, J.: Global asymptotic stability of bi-directional associative memory neural networks. Appl. Math. Comput. 142, 333–339 (2003)
 7. Cao, J., Wang, L.: Periodic oscillatory solution of bidirectional associative memory networks with delays. Phys. Rev. E 61(2), 1825–1828 (2000)
 8. Chen, A., Cao, J., Huang, L.: Exponential stability of BAM neural networks with transmission delays. Neurocomputing 57, 435–454 (2004)
 9. Liao, X., Wong, K.W.: Robust stability of interval bidirectional associative memory neural network with time delays. IEEE Trans. Syst. Man Cybern. 34(2), 1142–1154 (2004)
10. Li, C., Liao, X., Zhang, R.: Delay-dependent exponential stability analysis of bidirectional associative memory neural networks with time delay: an LMI approach. Chaos, Solitons Fractals 24, 1119–1134 (2005)
11. Yuan, K., Cao, J., Li, H.: Robust stability of switched CohenC Grossberg neural networks with mixed time-varying delays. IEEE Trans. Syst. Man Cybernet. 36(6), 1356–1363 (2006)
12. Cao, J., Huang, D., Qu, Y.: Global robust stability of delayed recurrent neural networks. Chaos, Solutions and Fractals 23, 221–229 (2005)
13. Singh, V.: Global robust stability of delayed neural networks, An LMI apporach. IEEE Trans. Circuits Syst. II, Exp. Brief. 53, 33–36 (2005)
14. Ozcan, N., Arik, S.: Global robust stability analysis of neural networks with multiple time delays. IEEE Trans. Circ. Syst. I 53(1), 166–176 (2006)
15. Boyd, S., Ghaoui, L.E., Feron, E., Balakrishnan, V.: Linear Matrix Inequalities in System and Control Theory. SIAM, Philadephia (1994)
16. Marcus, C.M., Westervelt, R.M.: Stability of analogy neural networks with delay. Phys. Rev. A 39, 347–359 (1989)
17. Shen, T., Zhang, Y.: Improved global robust stabilty criteria for delayed neural networks. IEEE Trans. Circuits Syst. II, Exp. Brief. 54, 715–719 (2007)

High Order Hopfield Network with Self-feedback to Solve Crossbar Switch Problem

Yuxin Ding, Li Dong, Bin Zhao, and Zhanjun Lu

Key Laboratory of Network Oriented Intelligent Computation,
Department of Computer Sciences and Technology,
Harbin Institute of Technology Shenzhen Graduate School, China
yxding@hitsz.edu.cn,
{dongli_hitsz,bzhao_hitsz,zhjlu_hitsz}@hotmail.com

Abstract. High order network has a higher store capacity and a faster convergence speed compared with the first order network. To improve the convergence speed of the energy function, in this paper a new kind of high order discrete neural network with self-feedback is proposed to solve crossbar switch problem. The construction method of the high order energy function for this problem is presented and the neural computing method is given. We also discuss the strategies for the network to escape from local minima. Compared with the first order Hopfield network, experimental results show the high order network with self-feedback has a quick convergence speed, its performance is better than the first order Hopfield network.

Keywords: Hopfield network, high order network, crossbar switch problem.

1 Introduction

The Hopfield network has been widely applied to solve different combinatorial problems, such as Traveling Salesman Problem [1], map coloring [2], maximum cut problems [3], bipartite subgraph problems [4], crossbar switch problem [5-7], and N-queens problem [8]. Due to the fact that high order neural network has stronger approximation property, faster convergence rate, greater storage capacity, and higher fault tolerance than the lower order neural networks, it has attracted great attention of many researchers over recent years. Although high-order networks have their own advantages, they are seldom applied to solve combinatorial optimization problems. The reason is that constructing a high order network that satisfies the stability criteria is more difficult than constructing the first order networks.

In the paper [5], a Hopfield-type high-order discrete neural network is proposed to solve crossbar switch problem. One interesting thing we found from [5] is that self-feedback mechanism can be introduced to the high order network without affecting its stability. In this paper we prove the stability of this kind of high order network with self-feedbacks, and show that the self-feedback of neurons is helpful to accelerate the convergence of the high order network.

2   Crossbar Switch Problem

2.1   Problem Description

An crossbar switch is a switch connecting a set of N inputs and N outputs where each input can be connected to each outputs as shown in Fig.1. When there is a request from the input to output be satisfied the Crosspoint switch will be closed. Some constraints for the problem are as follows: in each input line only one output line can be connected; in each output line only one input line can be connected. That is to say two or more requests coming simultaneously for the same output line only one request can be satisfied, and the other requests will be blocked.
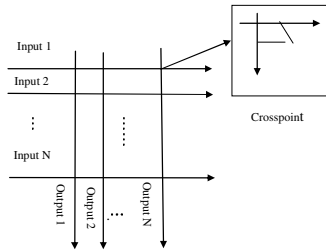
Fig. 1. *NxN* crossbar switch

A *NxN* crossbar switch can be represented by an *NxN* binary request matrix R [5]. Rows and columns of the matrix R correspond to the inputs set and outputs set, respectively. Each element in the matrix R has exactly two values: 0 and 1. $r_{ij}=1$ means there is a request from the ith input line to the jth output line; $r_{ij}=0$ expresses there is no request. The state of the switch can be represent by an *NxN* binary configuration matrix C, where $c_{ij}=1$ indicates that the request from the ith input line to the jth output line is satisfied. $c_{ij}=0$ indicates that the request is discarded. For proper operations of the switch, there should be at most one request being satisfied in each row and each column. The throughput of the switch is optimal when the matrix C, which is a subset of the matrix R, contains at most a "1" in each row/column, and has a maximum overlap with R.

2.2   Solving Crossbar Switch Problem Using High Order Discrete Network

Traditional Hopfield network (DHNN) has been applied to solve this problem, usually the energy function for this problem is defined as Eq.(1)[6]. It is a quadratic polynomial.

$$E = \frac{A}{2}\sum_{i=1}^{N}(\sum_{k=1}^{N}c_{ik}-1)^2 + \frac{B}{2}\sum_{j=1}^{N}(\sum_{k=1}^{N}c_{kj}-1)^2 \quad . \tag{1}$$

In Eq(1) c_{ij} is the output of neuron ij. The first term is zero, if and only if there is no more than one request being satisfied in each row. The second term is zero, if and only if there is no more than one request being satisfied in each column. The input u_{ij}

of neuron ij is derived by the partial derivative of the energy function with respect to the output of neuron ij, u_{ij} is shown as Eq(2).

$$u_{ij} = -A(\sum_{k=1}^{N} c_{ik} - 1) - B(\sum_{k=1}^{N} c_{kj} - 1) \ .$$

(2)

From (1) we can see that DHNN can only handle optimization problems whose energy functions can be expressed by a quadratic polynomial, if we want to deal with high-order problems, DHNN should be extend to high-order DHNN.

The paper [5] proposed a kind of high order discrete Hopfield network to solve crossbar switch problem. As mentioned in section 2.1, the state of a crossbar switch can be represented by a NxN binary configuration matrix C. c_{ij}=1 indicates that the request from i[th] input line to the jth output line is satisfied. c_{ij}=0 indicates that the request is discarded. Each c_{ij} is corresponding to r_{ij} in R. In crossbar problem we need to only consider the c_{ij} whose corresponding r_{ij} is 1(all the other c_{ij} equals 0). For expression convenience, all of c_{ij} are appeared in the following equations, however only c_{ij} whose corresponding r_{ij} is 1 is a variable, and all the other c_{ij} is a constant, whose value is 0. The constraint for the ith row is represented as form (3). Equation (3) equals zero, if and only if there is only one request be satisfied in i[th] row. In (3) the first term is a high order term. The constraint for all rows is represented as (4). Equation (4) equals zero, if and only if there is only one request be satisfied in each row. In the same way the column constraints for crossbar switch problem can be constructed. Equation (5) is the energy function for all columns. It equals zero if and only if there is only one request is satisfied in each column. The energy function for crossbar switch problem is the sum of (4) and (5), which is shown in Eq.(6). When Eq. (6) takes the minimum value 0, all c_{ij} are the solution of the problem.

$$\prod_{j=0}^{N-1}(1-c_{ij}) + \sum_{j=0}^{N-2} c_{ij} \sum_{k=j+1}^{N-1} c_{ik} = 0 \ .$$

(3)

$$E_1 = \sum_{i=0}^{N-1}(\prod_{j=0}^{N-1}(1-c_{ij})) + \sum_{i=0}^{N-1}\sum_{j=0}^{N-2} c_{ij} \sum_{k=j+1}^{N-1} c_{ik} \ .$$

(4)

$$E_2 = \sum_{j=0}^{N-1}(\prod_{i=0}^{N-1}(1-c_{ij})) + \sum_{j=0}^{N-1}\sum_{i=0}^{N-2} c_{ij} \sum_{k=i+1}^{N-1} c_{kj} \ .$$

(5)

$$E = E_1 + E_2 \ .$$

(6)

The neural computing algorithm working in serial mode is shown as following:

Algorithm 1

Step 1: $t = 0$

Step 2: initialize all c_{ij} *(0)* to 1 for r_{ij} =1, and all c_{ij} *(0)* to 0 for r_{ij} =0 ($i, j = 0,1,...N-1$)

Step 3: For $i = 0$ to $N-1$
For $j = 0$ to $N-1$ {

    If $r_{ij} = 0$ continue;

    $d_{ij}(t) = -\partial E / \partial c_{ij}(t)$

    $c_{ij}(t+1) = f_h(d_{ij}(t))$

    let all $c_{mn}(t+1) = c_{mn}(t)$, $(m \neq i$ or $n \neq j)$

    $t = t+1$ }

Step 4: if all $c_{ij}(t) = c_{ij}(t-1)$ return all $c_{ij}(t)$ else goto step 3

In the above algorithm $f_h(y)$ is a hysteretic threshold function. If $y<0$, then $f_h(y)=0$; if $y>0$, then $f_h(y)=1$; when $y=0$, $f_h(y)$ keeps unchanged (in fact, this function can be seen as a special case of the sigmoid function). E is the energy function defined in Eq.(6). E is a polynomial function with respect to c_{ij}, therefore it is easy to calculate the derivative term d_{ij}. The description of the high order energy function and the convergence of the high order network are discussed in detail in [5] [8].

3   Introducing Self-feedbacks to the High Order Network

The neural network model for Algorithm 1 is a single layer feedback neural network. The paper [5] proved that the energy of the network defined by Eq. (6) decreases or remains the same after each unit updates, and the network will eventually converge to a local minimum of the energy function(shown as form (7), for the detailed proof of Eq.(7), please refer to the paper[5]).

$$\Delta E = E(t+1) - E(t) = (c_{ij}(t+1) - c_{ij}(t))\partial E / \partial c_{ij} \leq 0 \qquad (7)$$

Considering the characteristics of the energy function E, we can add a penalty to the energy function without affecting the convergence of the high order network.   The energy function with penalty is defined as Eq.(8). We can also prove that the value of the energy function (8) decreases or remains the same after each unit update, and the network will eventually converge to a local minimum of the energy function.

$$E' = E + k \sum_{i,j=0}^{N-1}(1-c_{ij})c_{ij} \qquad (8)$$

Proof: we have

$$\Delta E' = E'(t+1) - E'(t) = (c_{ij}(t+1) - c_{ij}(t))\partial E'/\partial c_{ij}(t) - k(c_{ij}(t+1) - c_{ij}(t))^2$$

If $\partial E'/\partial c_{ij} < 0$, then $c_{ij}(t+1) = 1$, we get $\Delta E' \leq 0$

If $\partial E'/\partial c_{ij} \geq 0$, then $c_{ij}(t+1) = 0$, we get $\Delta E' \leq 0$

This completes the proof.

    In the above proof we can see that except a non-positive derivative term, $\Delta E'$ contains another non-positive term $-k(c_{ij}(t+1)-c_{ij}(t))^2$. If E and E' are initialized using the same c_{ij}, E' is equal to E. We can adjust the value of k to control the convergence speed of the energy function. Usually a big value of k is helpful to decrease the value of the energy function fast. From the next section we can see that adding penalty is equativelent to introducing self-feedback to each neuron.

Adding the penalty term can improve the convergence speed, however it cannot guarantee to reach the global minimum of the energy function. For the energy function, its minimum value is zero, and all of its term is non-negative, thus every term must be equal to 0. Taking advantage of this heuristic information, we can adopt the following method to escape from local minima: select some variables c_{ij} randomly from non-zero terms and reassign values to them, and then repeat the iteration process. This procedure is also called a disturbance. In this paper in order to keep the energy value maintained at a relatively low level, we take the smallest disturbance, randomly select only one non-zero term and randomly change the values of each variable, and then repeat the iteration process.

Hopfield-type network is easy to fall into local minimum. There are two approaches usually adopted to escape from local minima: stochastic approach and deterministic approach. The deterministic approaches include the "divide and conquer" method [9], the "rock and roll" perturbation method [10], and neurons' competitive learning method [12]. Stochastic approaches include genetic algorithm [13], annealing theory [11], particle swarm optimization [3], ant colony algorithm [14], and so on. Although theoretically stochastic approaches can reach the global optimum, practically it is very difficult to achieve. It not only takes long running time, but also very difficult to determine the termination conditions. All methods are suitable for the high-order gradient descent network. Our target is to compare the difference in performance caused by the topologies of the high order network and the lower order network, so in this paper we do not use any global optimization strategies discussed on above.

From the energy function (8), we can see that in the energy function E' each variable c_{ij} has two forms, one is c_{ij} and another is 1- c_{ij}. According to [5], we expand the structure of neurons, and each neuron has two outputs: one is the positive output c_{ij}, and the other is the negative output1- c_{ij}. From the topological point of view, the penalty can be translated into two self-feedbacks to each neuron, one is called positive self-feedback, and the other is called negative self-feedback. The weight of positive self-feedback is negative, its value is -k; the weight of negative self-feedback is positive, its value is k (k is a positive integer that is the coefficients of the penalty). The topological structure of the network is shown in Fig.2.
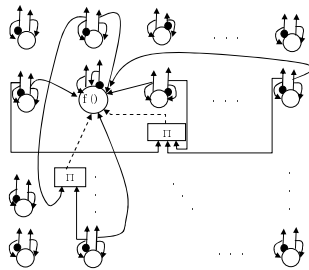
Fig. 2. Neural network topology for crossbar switch problem

According to Eq.(8), the input of a neuron c_{ij} is the sum of the following five parts:

1) The product of the negative outputs of neurons that are in the same row as c_{ij};

2) The negative of the sum of the positive outputs of neurons that are in the same row as c_{ij};

3) The product of the negative outputs of neurons that are in the same column as c_{ij};

4) The negative of the sum of the positive outputs of neurons that are in the same column as c_{ij};

5) The positive self-feedback and negative self-feedback from each neuron.

The first and third parts represent high order weights. Take c_{11} as an example, its input is shown in Fig.2. The neural network is represented as an NxN neuron matrix. Each neuron has two outputs; the black dot represents the negative output. Π is a multiplier. c_{11} has two high order weights represented by the dot line, and their weight value is 1.

4   Simulation Results

The first experiment is to illustrate that the self-feedback can accelerate the convergence speed of the energy function. From the proof in section 3 we can see that a bigger value of k is helpful to make the energy function converge fast. However, when $c_{ij}(t+1)$ is not equal to $c_{ij}(t)$, the reduced value of $\Delta E'$ is greater than k, since the minimum value of $\Delta E'$ is 0, k cannot be too large, overlarged k will overshoot the minima, this makes the value of the energy function drops quickly to a higher value, then does not continue to fall. We take 20×20 crossbar switch problem as an example to interpret this problem. We assign different values to the penalty coefficient k, the curves of energy functions corresponding to different k are shown in Fig.3. From Fig.3 we can see the value of the energy function decreases fast with the increase of k. This means the convergence speed of E is increased with the increase of k. However, for k=20 the energy decreases to a certain small value, and then stops. To avoid it, k should be decreased gradually with the increase of iteration number, when $\Delta E'$ is close to 0, k should be set to 0.
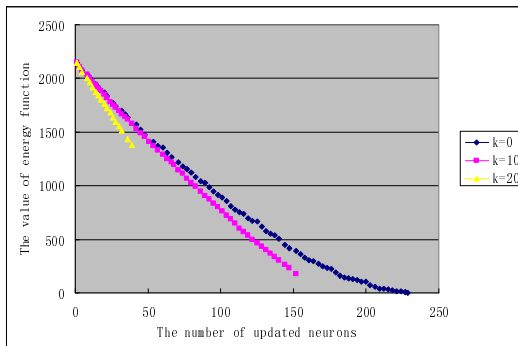
Fig. 3. Energy curves for different penalty coefficients for 20×20 crossbar switch problem

To test the performance of the high order network, we use it to solve different size of crossbar switch problems. In the experiments the size N ranges from 20 to 200. For each $N \times N$ crossbar switch problem, we simulated it 100 times and in each simulation the request matrix is randomly initialized. For a simulation if a solution is found, the simulation is called a convergent simulation. In order to test the high order network's performance, we compare it with the traditional discrete Hopfield neural network (DHNN) [15]. The energy function of DHNN is defined as Equation (5). The parameters A and B are equal to 1. In our algorithm the penalty coefficient is set to 1. In both algorithms neurons are selected in a fixed order to be updated, when all the neurons are checked one time, we call this is one step. Each simulation is terminated if a solution is found or the iteration step exceeds the maximum iteration step 100. The performance of different algorithms is evaluated by the following 2 criteria:

(1) Convergence rate: the ratio of the number of convergent simulations to the total number of simulations.

(2) Average iteration steps: the average iteration steps required for each simulation.

Table 1. Performance comparison for different size crossbar switch problems

N	Avg. Step		Converge Rate (%)	
	DHNN	HHNN	DHNN	HHNN
20	13.5	3.9	100	100
30	17.4	3.8	100	100
50	29.2	3.7	100	100
80	45.4	3.3	100	100
100	57.8	3.6	100	100
200	101.6	4.1	100	100

The simulation results are shown in table 1. In table 1 the high order Hopfield network with self-feedback is denoted as HHNN. From table 1 we can see that two types of neural networks have 100% convergence rate. However, the performance of HHNN with self-feedbacks is much better than DHNN. In average HHNN needs fewer steps than DHNN to get a solution, especially for a larger N the improvement of the performance is significant. It is because the high order network structure accelerates the convergence speed of the energy function. Furthermore, we can see that the number of steps to get a solution for HHNN is almost independent of the size of crossbar switches, while the iteration steps of DHNN increases greatly with the increase of the size of the problem.

5　Summary

In this paper we propose a high order network with self-feedback to solve crossbar switch problem. We prove the stability of the high order neural network, and interpret why self-feedback can accelerate the convergence speed of the high order network. The simulation results show higher order network with self-feedbacks has a quicker convergence speed than the first order network.

References

1. Hopfield, J.J., Tank, D.W.: Neural computation of decisions in optimization problems. Biol. Cybern. 52, 141–152 (1985)
2. Galán-Marín, G., et al.: A Study into the Improvement of Binary Hopfield Networks for Map Coloring. In: Beliczynski, B., Dzielinski, A., Iwanowski, M., Ribeiro, B. (eds.) ICANNGA 2007. LNCS, vol. 4432, pp. 98–106. Springer, Heidelberg (2007)
3. Wang, J.: A Memetic Algorithm with Genetic Particle Swarm Optimization and Neural Network for Maximum Cut Problems. In: Li, K., Fei, M., Irwin, G.W., Ma, S. (eds.) LSMS 2007. LNCS, vol. 4688, pp. 297–306. Springer, Heidelberg (2007)
4. Wang, J., Tang, Z.: An improved optimal competitive Hopfield network for bipartite subgraph problems. Neurocomputing 61, 413–419 (2004)
5. Ding, Y., Dong, L., Wang, L., Wu, G.: A High Order Neural Network to Solve Crossbar Switch Problem. In: Wong, K.W., Mendis, B.S.U., Bouzerdoum, A. (eds.) ICONIP 2010, part II. LNCS, vol. 6444, pp. 692–699. Springer, Heidelberg (2010)
6. Xia, G.: A binary Hopfield neural network with hysteresis for large crossbar packet-switches. Neurocomputing 67, 417–425 (2005)
7. Troudet, T.P., Walters, S.M.: Neural Network Architecture for Crossbar Switch Control. IEEE Trans. Circuits Syst. 38, 42–56 (1991)
8. Cheung, K., Lee, T.: On the Convergence of Neural Network for higher Order Programming. In: 1993 International Joint Conference on Neural Networks, pp. 1507–1511. IEEE Press, New York (1993)
9. Foo, Y.P.S., Szu, H.: Solving Large-Scale Optimization Problems by Divide-and-Conquer Neural Networks. In: 1989 International Joint Conference on Neural Networks, pp. 507–511. IEEE Press, New York (1989)
10. Lo, J.T.-H.: A New Approach to Global Optimization and its Applications to Neural Networks. Neural Networks 2, 367–373 (1992)
11. Wang, L.P., Li, S., Tian, F.Y., Fu, X.J.: A Noisy Chaotic Neural Network for Solving Combinatorial Optimization Problems: Stochastic Chaotic Simulated Annealing. IEEE Trans. on Sys. Man, Cybern. Part B 34, 2119–2125 (2004)
12. Amartur, S.C., Piraino, D., Takefuji, Y.: Optimization Neural Networks for the Segmentation of Magnetic Resonance Images. IEEE Trans. on Medical Imaging 11, 215–220 (1992)
13. Salcedo-Sanz, S., Yao, X.: A hybrid Hopfield network-genetic algorithm approach for the terminal assignment problem. IEEE Trans. on Systems, Man and Cybernetics 34, 2343–2353 (2004)
14. Li, Y.: An Improvement to Ant Colony Optimization Heuristic. In: Sun, F., Zhang, J., Tan, Y., Cao, J., Yu, W. (eds.) ISNN 2008, Part I. LNCS, vol. 5263, pp. 816–825. Springer, Heidelberg (2008)
15. Hopfield, J.J.: Neural networks and physical systems with emergent collective computational abilities. Proceedings of the National Academy of Sciences of the USA 79(8), 2554–2558 (1982)

Use of a Sparse Structure to Improve Learning Performance of Recurrent Neural Networks

Hiromitsu Awano[1], Shun Nishide[1], Hiroaki Arie[2], Jun Tani[2],
Toru Takahashi[1], Hiroshi G. Okuno[1], and Tetsuya Ogata[1]

[1] Kyoto University, Graduate School of Informatics,
Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, Japan
[2] RIKEN, Brain Science Institute,
2-1 Hirosawa, Wako City, Saitama 351-0198, Japan

Abstract. The objective of our study is to find out how a sparse structure affects the performance of a recurrent neural network (RNN). Only a few existing studies have dealt with the sparse structure of RNN with learning like Back Propagation Through Time (BPTT). In this paper, we propose a RNN with sparse connection and BPTT called Multiple time scale RNN (MTRNN). Then, we investigated how sparse connection affects generalization performance and noise robustness. In the experiments using data composed of alphabetic sequences, the MTRNN showed the best generalization performance when the connection rate was 40%. We also measured sparseness of neural activity and found out that sparseness of neural activity corresponds to generalization performance. These results means that sparse connection improved learning performance and sparseness of neural activity would be used as metrics of generalization performance.

Keywords: Recurrent Neural Networks, Sparse Structure, Sparse Coding.

1 Introduction

Several studies have suggested that neurons encode information by using a small number of active neurons at any given point in time. This phenomenon is called "sparse coding". According to Olshausen [1], sparse coding has four advantages: it increases storage capacity, makes complicated signals more explicit, makes reading out information coded in a network easier, and saves energy. There have been many studies on sparse coding theory. For example, Waydo et al. showed that neurons in the human medial temporal lobe are remarkably selective to sensory inputs [2]. Smith et al. showed that a process like independent component analysis is used in the auditory nervous system [3]. Vinje and Gallant showed that neurons in the V1 area of the brain exhibit a sparse response when stimulated with image sequences [4]. While these studies have shown the advantage of sparse coding in sensory neurons, computational studies on associative memory have shown that sparse representation of information increases memory capacity and fault tolerance [5] [6].

Several previous studies on recurrent neural networks used a sparse structure. The echo-state network (ESN), for example, uses a "reservoir" to store chaotic sequences [7]. It can store a huge variety of such sequences, because the neurons in the reservoir are sparsely connected. While the connection weight of ESN was fixed, the self-organizing recurrent neural network (SORN) used Hebb's learning rule to train the weight [8]. These previous recurrent neural networks using a sparse structure cannot learn complex sequences such as robot's motor value because the connection weight is fixed or trained using local learning (e.g. Hebb's learning).

In this paper, we investigate how sparse structure affects performance of recurrent neural network (RNN) trained by back-propagation through time (BPTT) algorithm [9]. The BPTT algorithm enables the RNN to expand its ability for training quite complicated data. However, there are few studies dealing with the sparseness of RNN with BPTT. We treat the learning of "language structure" ("word" and "grammar") as the complicated task, and use the MTRNN (Multiple Timescale RNN [10]) for the training. The relationships between sparse connection and sparse coding in the network are investigated by introducing a sparseness measure.

2   Target Model

We used the multiple timescale recurrent neural network (MTRNN) as the target model. We introduced a sparse structure into this model and investigated its performance.

2.1   MTRNN

MTRNN, shown in Fig. 1 is a variant of the Jordan RNN with two layers. One layer acts as a predictor that inputs the current state, $IO(t)$, and, outputs the next state, $IO(t+1)$, where $IO(t)$ represents the input value. The context nodes of MTRNN are composed of three neuron groups with different time constants: fast context nodes (C_F), slow context nodes (C_S) and parametric slow context nodes (PC_S). The time constant of the input nodes is the smallest out of all the nodes. It increases in the order: C_F, C_S, PC_S. Each node in the input layer is connected to one in the output layer, with the exception of the $IO(t)$ nodes to the C_S and PC_S nodes and the C_S and PC_S nodes to the $IO(t)$ nodes. The differences in the time constants lead to differences in the firing speeds of the context nodes. Therefore, the C_F nodes represent the primitives of sequential data while the C_S nodes represent the sequence of these primitives (Fig. 2).

MTRNN can generate sequence data by connecting output nodes to its input nodes. It can also calculate the initial state of the parametric neuron corresponding to the input data. The space of the initial state is self organized depending on the input data. Thus, MTRNN can be used as a prediction and recognition unit. Compared with Recurrent Neural Network with Parametric Bias (RN-NPB), MTRNN has the advantage in terms of complexity and the ability to learn sequences.

Training of MTRNN is done using the back propagation through time algorithm, which consists of forward calculation and weight updating.

First, the outputs of the neurons are calculated through forward calculation. The internal value of the ith neuron, $u_i(t)$, at step t is given by

$$u_i(t) = \left(1 + \frac{1}{\tau_i}\right) u_i(t-1) + \frac{1}{\tau_i} \left[\sum_{j \in N} w_{ij} x_j(t-1)\right], \qquad (1)$$

where τ_i is the time constant of the ith neuron, w_{ij} is the weight value from the jth input neuron to the ith output neuron, and $x_j(t)$ is the input value. The output of the ith neuron, $y_i(t)$, is calculated by applying the sigmoid function.

$$y_i(t) = \frac{1}{1 + exp\left(-u_i(t)\right)}. \qquad (2)$$

The input value, $x_i(t)$, is calculated using

$$x_i(t) = \begin{cases} 0.5 \times y_i(t-1) + 0.5 \times T_i(t) & i \in IO, \\ y_i(t-1) & otherwise \end{cases} \qquad (3)$$

where $T_i(t)$ is the teacher signal for the ith neuron at step t. The input values, $x_i(t)$, are calculated using the output of the previous step, $y_i(t-1)$, and feeding back the teacher signals. This process prevents the training error from increasing. The outputs of the context nodes ofp the previous step, $(t-1)$, $C_F(t-1)$, $C_S(t-1)$, and $PC_S(t-1)$, are directly fed into the input of the context nodes of the current step, t, $C_F(t)$, $C_S(t)$, and $PC_S(t)$.

The weights are updated using the outputs calculated in forward calculation and the training error, defined as

$$E = \frac{1}{2} \sum_t \sum_{n \in IO} \left(y_i(t-1) - T_i(t)\right)^2. \qquad (4)$$

The weight from the jth input to the ith output is updated using the derivative of the training error, $\frac{\partial E}{\partial w_{ij}}$:

$$w_{ij}(n+1) = w_{ij}(n) - \alpha \frac{\partial E}{\partial w_{ij}}. \qquad (5)$$

The initial PC_S value, $PC_S(0)$, is also updated along with the weight values by back propagating the errors to the initial step.

With the trained MTRNN, sequences can be recovered by using the PC_S value. The predicted value of step 1 $(IO(1))$ can be calculated by inputting the initial PC_S value $(PC_S(0))$, the C_S value, the C_F value (which is fixed

to zero), and the initial value of $IO(0)$. The whole sequence can be recovered by recursively inputting the output back into the input. This process is called "closed-loop calculation". In contrast, the process in which the observed values for each step are input is called "open-loop calculation".

2.2 Introduction of Sparse Structure

We represent sparse connections by setting the weights of some connections to zero.

These weights are clamped to zero and the other weights are updated during training phase.

These connections are randomly selected in accordance with the connection rate. For example, if the connection rate is 20%, the weights for 80% of the connections, randomly selected, are set to zero. Initial weights of remaining weights were set to small random numbers from interval (x, y).
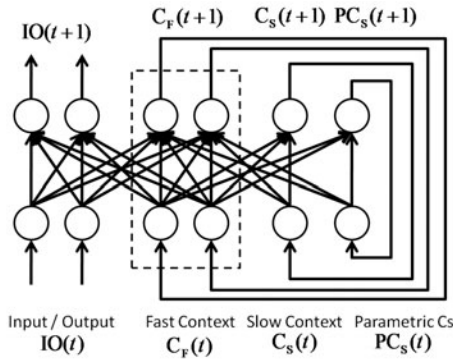
Fig. 1. Structure of MTRNN

3 Evaluation

We conducted two experiments to evaluate the performance of the proposed model. In the first, we evaluated a generalization performance and robustness to noisy data relative to the connection rate. In the second, we introduced a sparseness measure and investigated the sparseness of information coded in the network.

3.1 Performance Evaluation

We used data comprising alphabetic sequences to evaluate the performance of the proposed model [11]. Each neuron in the input and output groups corresponded to a letter, comma, period, and space. A neuron fired when the corresponding element appeared. Figure 3 shows example data used for the evaluation. The left vertical axis corresponds to a neuron, and the horizontal axis is the time scale. The color bar on the right shows neuron output: red means that the neuron is firing and white

means that it is not. The example data in Fig. 3 represents "walk slowly". The parameter settings for MTRNN are shown in Table 1. The neurons in the C_F nodes (connection inside dotted line in Fig. 1) are sparsely connected. The connection rate was incrementally increased from 20% to 100% in steps of 20%. The connection weights were initialized using 20 sets of random values.

Generalization Capability. Generalization means adapting to unknown data. We examined the ability of MTRNN to reproduce sequences that were not used for training. First, the evaluation data was input to the input node of MTRNN. Using forward calculation, MTRNN output the predicted values. The difference between the predicted values and the actual data were calculated, and the initial value of the neuron in the parametric group was updated using the difference (error signal). These values were used as the initial values for the input and parametric nodes. The whole sequence was generated (closed-loop calculation) by recursively inputting the predicted value to the input node. We calculated the square error between the generated sequence and the actual sequence.

Robustness to Noisy Data. We examine the ability of MTRNN to reproduce clean data when noisy data was used to calculate the initial values of the parametric nodes. Noisy data was created by replacing an element in the original data. The initial values of the parametric nodes were calculated, and the whole sequence was recovered by closed-loop calculation. The square error between the recovered sequence and the actual sequence without noise was calculated as the metric.
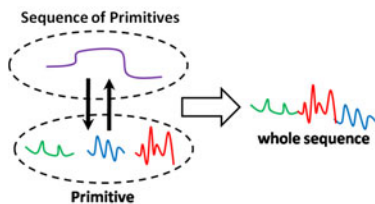
Fig. 2. Hierarchal representation of sequence coded in MTRNN
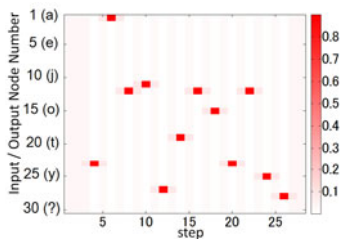
Fig. 3. Data used for evaluation

3.2 Measuring Sparseness

We measured the sparseness of neural output when the network was generating sequences with the MTRNN parameters set to the same values as in the first experiment. Methods for measuring the sparseness of the neural response include the use of the kurtosis of the neural output [12]. In this study, we used a simpler measure: "activity sparseness", which was proposed by Willmore and Tolhurst [13]. This measure was directly derived from the basic idea of sparse coding: 'how few neurons are active at any given time'. We set a threshold value for

Table 1. MTRNN Parameter Settings

No. of Input Nodes	30
No. of Fast Context Nodes	200
No. of Slow Context Nodes	6
No. of Parametric Slow Context Nodes	6

the response of each neuron. The neurons with a response greater than the threshold were considered "active". The others were considered "inactive". The activity sparseness at any given time is given by the number of neurons that are "inactive" divided by the number of whole neurons. The time average activity sparseness is given by

$$S_{activity} = \frac{1}{T} \sum_t \frac{the \ number \ of \ inactive \ neurons}{the \ number \ of \ whole \ neurons}, \tag{6}$$

Where T indicates the number of whole steps.

The threshold value was set to 0.5 in this study because we used a sigmoid function to activate the neurons in the C_F node. The internal value of a neuron was zero when the output value (after calculating the sigmoid function) was 0.5. This means that the neurons that were "active" had a positive internal value and the neurons that were "inactive" had a negative internal value.

4   Results

4.1   Performance Evaluation

The reproduction accuracy for untrained data is shown in Fig. 4. Each bar indicates the average reproduction accuracy for the top ten patterns; the error bars represent the maximum and minimum reproduction accuracy of the patterns. The MTTNN showed the best performance when the connection rate was 40%. The reproduction accuracy for noisy data is shown in Fig. 5. As with Fig. 4, each bar indicates the average accuracy and the error bars represent the maximum and minimum accuracy. The MTRNN showed the best performance when the connection rate was 60%. We also calculated two-sided p-value for each experiment with level of significance set to 0.05. The results are shown in Table 2, 3. From Table 2, it can be noted that the reproduction error for connection rate 40% is significantly low compared to other connection rate for untrained data.

4.2   Measuring Sparseness

The value of activity sparseness is shown in Fig. 6. Each bar indicates the average activity sparseness for 20 patterns. The error bars represent the maximum and minimum value of activity sparseness. The MTRNN had the maximum value of

activity sparseness when the connection rate was 40 %. As with previous subsection, we calculated two-sided p-value between each connection rate. The results are shown in Table 4. The level of significance was set to same as previous subsection. From Table 4, it can be noted that the activity sparseness for connection rate 40% is significantly high compared to connection rates 20%, 80% and 100%.
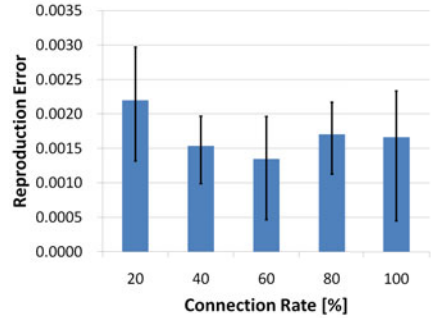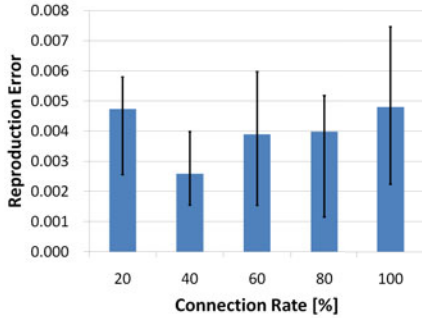
Fig. 4. Reproduction error for untrained data

Fig. 5. Reproduction error for noisy data
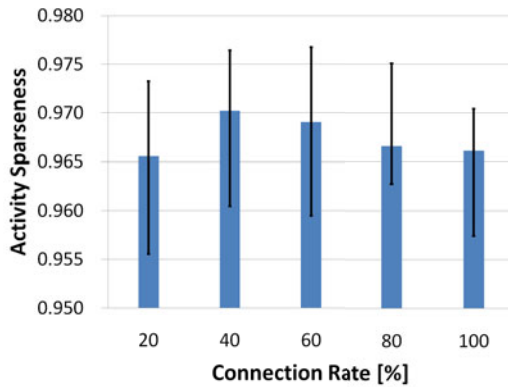
Fig. 6. Activity sparseness

Table 2. p-value of reproduction error for untrained data: A combination which has significant difference is denoted by an asterisk

connection rate [%]	40	60	80	100
20	0.0299*	0.814	0.552	0.196
40		0.0401*	0.0267*	0.00445*
60			0.731	0.327
80				0.548

Table 3. p-value of reproduction error for noisy data: A combination which has significant difference is denoted by an asterisk

connection rate [%]	40	60	80	100
20	0.00698*	0.00338*	0.108	0.480
40		0.800	0.249	0.172
60			0.161	0.124
80				0.590

Table 4. p-value of activity sparseness: A combination which has significant difference is denoted by an asterisk

connection rate [%]	40	60	80	100
20	0.00453*	0.0836	0.755	0.914
40		0.275	0.00266*	0.00156*
60			0.0954	0.0650
80				0.811

5   Discussion

In the evaluation of generalization performance, the MTRNN performed best when the connection rate was 40%. A connection rate of less than 40% resulted in worse performance. An MTRNN with highly sparse connections has better performance for learning sequences because there are fewer collisions between sequences. However, in our experiment, it showed lower generalization performance. An MTRNN with dense connections, on the other hand, has a larger parameter space with many local solutions. The difference between the maximum and minimum reproduction accuracy tends to become smaller as the degree of sparseness increases. These findings mean that appropriate selection of the connection rate is required to avoid a local solution while retaining generalization capability.

   In the evaluation of activity sparseness, the MTRNN showed the maximum sparseness when the connection rate was 40%. The value of activity sparseness corresponds to generalization performance. An MTRNN with high activity sparseness tends to have better generalization performance. This means that we can judge whether a primitive is well organized by doing a simple calculation: counting the number of "inactive neuron".

6   Conclusion

We introduced sparse connection into an MTRNN model trained by BPTT algorithm, and investigated how sparse connection affects generalization performance and noise robustness. In an experiment using data composed of alphabetic sequences, the MTRNN had the best generalization performance when the connection rate was 40% and had the highest robustness when the rate was 60%.

Future work including testing using data with a different structure (e.g. the motor value of a robot). It also includes analyzing the structure of MTRNN and investigating its robustness after additional training. We speculate that increasing connection sparseness will reduce collisions between the learned sequences and the additional sequences, leading to robustness after additional training.

References

1. Olshausen, B.A., Field, D.J.: Sparse coding of sensory inputs. Current Opinion in Neurobiology 14(4), 481–487 (2004)
2. Waydo, S., Kraskov, A., Quian Quiroga, R., Fried, I., Koch, C.: Sparse representation in the human medial temporal lobe. J. Neurosci. 26(40), 10232–10234 (2006)
3. Smith, E., Lewicki, M.: Efficient auditory coding. Nature 439(7079), 978–982 (2006)
4. Vinje, W.E., Gallant, J.L.: Sparse Coding and Decorrelation in Primary Visual Cortex During Natural Vision. Science 287(5456), 1273–1276 (2000)
5. Kanerva, P.: Sparse distributed memory and related models, pp. 50–76. Oxford University Press, Inc. (1993)
6. Palm, G., Sommer, F.: Associative data storage and retrieval in neural networks. Models of Neural Networks III, 79–118 (1996)
7. Jaeger, H., Haas, H.: Harnessing Nonlinearity: Predicting Chaotic Systems and Saving Energy in Wireless Communication. Science 304(5667), 78–80 (2004)
8. Andreea, L., Pipa Gordon, T.J.: SORN: a self-organizing recurrent neural network. Front. Comput. Neurosci. 3(23) (2009), doi:10.3389/neuro.10.023
9. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning internal representations by error propagation, pp. 318–362. MIT Press (1986)
10. Yamashita, Y., Tani, J.: Emergence of functional hierarchy in a multiple timescale neural network model: A humanoid robot experiment. PLoS Comput. Biol. 4(11), e1000220 (2008)
11. Hinoshita, W., Arie, H., Tani, J., Okuno, H.G., Ogata, T.: Emergence of hierarchical structure mirroring linguistic composition in a recurrent neural network. Neural Networks (in Press, 2011), doi:10.1016/j.neunet.2010.12.006
12. The 'independent components' of natural scenes are edge filters. Vision Research 37, 3327–3338 (1997)
13. Willmore, B., Tolhurst, D.: Characterizing the sparseness of neural codes. Network: Computation in Neural Systems 12(3), 255–270 (2001)

Recall Time Reduction of a Morphological Associative Memory Employing a Reverse Recall

Hidetaka Harada and Tsutomu Miki

Graduate School of Life Science and Systems Engineering,
Kyushu Institute of Technology, Kitakyushu 808-0196, Japan
`harada-hidetaka@edu.brain.kyutech.ac.jp,`
`miki@brain.kyutech.ac.jp`

Abstract. As one of associative memory models, a morphological associative memory (MAM) has been proposed by Ritter. The model has advantages of large memory capacity and high perfect recall rate in comparison with other associative memory models. Unfortunately, the conventional MAM has a problem that it cannot recall the correct pattern for a pattern completely included with other stored patterns. To overcome the problem, we proposed a MAM employing a reverse recall. However, this model needs additional calculations for the reverse recall. The extra recall time increases as the number of included patterns increases. In this paper, as one of the solutions, we propose a MAM employing a simplified reverse recall. The extra recall time of the proposed model can be reduced by simplifying the calculation of the reverse recall for binary patterns. We confirm the validity of the proposed method by evaluating the recall time on hetero-association experiments.

Keywords: morphological associative memory, reverse recall, pattern with inclusive relation, recall time.

1 Introduction

Human beings can recall reasonable information from information obtained from sensory organs even if it is incomplete. This kind of function is called associative memory which is one of the important brain functions. The associative memory models have been proposed from early in 80's [1-4]. As the most popular associative memory model, Hopfield network [1] is well known. The model is used as not only the associative memory but also an optimization tool. However, it is also known that Hopfield network has drawbacks of low memory capacity in contrast to the number of memory units and instability in recall caused by the local minimum.

On the other hand, a morphological associative memory (MAM) has been proposed by Ritter [5]. The model has advantages of large memory capacity and high perfect recall rate in comparison with other associative memory models such as Hopfield network. Unfortunately, Ritter's MAM has a drawback that the design of the kernel image used for an association becomes difficult when the number of stored patterns increases. Moreover, the perfect recall rate of the MAM without the kernel image is inferior to the MAM with the kernel image. To overcome the problem,

several effective design methods of the kernel image have been proposed [6-8]. However, those models have a problem that the perfect recall cannot be achieved for pattern including a corrupted kernel image, or the stored patterns having redundancy bits are needed for the kernel design method. In our previous work, we proposed a MAM using an effective kernel design method that employs the stored pattern independent kernel image [9]. The model facilitated the design of the kernel image and improved the perfect recall rate. Unfortunately, as well as conventional MAMs, the model also has a problem that the correct pattern cannot be determined for a pattern completely included with other stored pattern (e.g., "C and G", "E and F"). As the number of the stored patterns increases, the existence of inclusive patterns becomes serious. Therefore, we proposed a MAM employing a reverse recall method [10]. In this model, the reverse recall method is introduced into the MAM using the stored pattern independent kernel image. In the reverse recall method, the candidates of an input pattern can be estimated by the reverse recall, and then the plausible recall pattern can be determined by comparing the input pattern with the candidates obtained by the reverse recall. However, the MAM employing the reverse recall needs additional calculations for the reverse recall method. When the number of included patterns increases, the extra recall time increases for the reverse recall.

In the case of binary patterns, the calculation of the reverse recall can be simplified. By simplifying the calculation, the stored pattern can be reversely recalled in one-shot. The MAM, even if the stored patterns are limited to binary, has been used for many applications; ex. the model is applied to hyperspectral image analysis, image recognition and binary image processing, etc. [11-13]. Therefore, in this paper, as one of the solutions, we propose a MAM employing the simplified reverse recall method for reducing the extra recall time. In order to confirm the validity of the proposed method, we evaluate the recall time and the perfect recall rate by hetero-association experiments.

2   Morphological Associative Memory: MAM

2.1   Ritter's MAM

Ritter proposed the MAM [5] that has two-stage recall process using memory matrices "M" and "W" in the stages, as shown in Fig.1. In the recall process, a kernel image is used as an index for an association. The information of the stored patterns is stored into the memory matrices "M" and "W". Here, let $(X^1, Y^1), \cdots, (X^R, Y^R)$ be R stored pattern pairs with $X^r = (x_1^r, \cdots, x_n^r)$ and $Y^r = (y_1^r, \cdots, y_m^r)$. The memory matrices "$M$" and "$W$" are given as;

$$m_{ij} = \bigvee_{r=1}^{R} (z_i^r - z_j^r), \tag{1}$$

$$w_{ij} = \bigwedge_{r=1}^{R} (y_i^r - z_j^r), \tag{2}$$

where R is the total number of the pattern pairs. m_{ij} and w_{ij} are (i, j)-th unit of memory matrices "M" and "W", respectively. The symbols $\vee$ and $\wedge$ denote

maximum and minimum operators, respectively. The kernel image Z^r is used to recall the stored pattern Y^r and consists of partial units of the stored pattern X^r. The output pattern Y^r corresponding to the input pattern X^r is obtained as follows;

$$z_i^r = \bigwedge_{j=1}^{n} \left(m_{ij} + x_j^r \right) \qquad i = 1, \cdots, n, \tag{3}$$

$$y_i^r = \bigvee_{j=1}^{n} \left(w_{ij} + z_j^r \right) \qquad i = 1, \cdots, m. \tag{4}$$
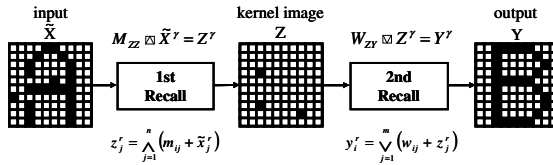
Fig. 1. Recall process of Ritter's morphological associative memory [5]

2.2   MAM Using an Effective Kernel Design Method and a Reverse Recall

In Ritter's MAM, the design of the kernel image becomes hard as the number of stored patterns increases because the kernel image consists of partial units of the stored pattern and it cannot overlap each other. To overcome this problem, we proposed the MAM using the stored pattern independent kernel image [9]. In this kernel design method, only one element of the kernel image is '1' and other elements '0'. Fig.2 shows the recall process of the MAM using the stored pattern independent kernel image. This model has advantages that the design of the kernel image is facilitated and the perfect recall rate is improved.
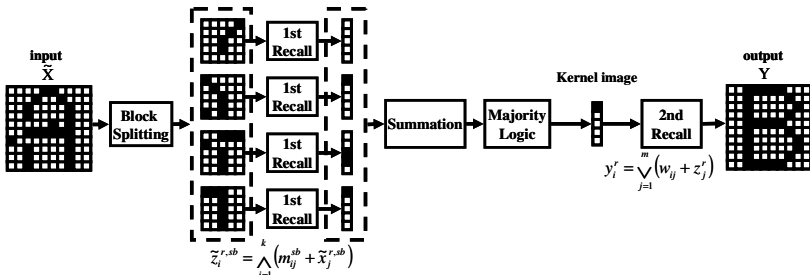
Fig. 2. Recall process of the MAM using the stored pattern independent kernel image [9]

However, as well as conventional MAMs, the MAM using the stored pattern independent kernel image has a problem that the correct pattern cannot be recalled if a pattern is completely included with other stored patterns (a pattern in the pair is called "inclusion pattern"), as shown in Fig.3.
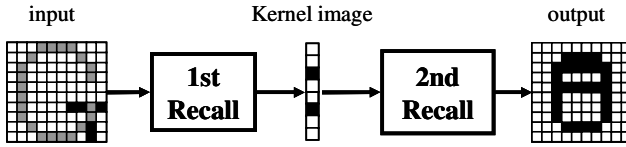
Fig. 3. Recall process of the MAM using the stored pattern independent kernel image for the inclusion pattern in hetero-association. ■ denotes "C" completely included with "G".

As one of the solutions, we introduced the reverse recall method into the MAM using the independent kernel image [10]. The recall process of the MAM employing the reverse recall, as shown in Fig.4, executes the following steps;

step1.   the kernel image is determined in the 1st recall, as same as the MAM using the stored pattern independent kernel image,

step2.   the overlapped kernel image is separated to individual kernel images,

step3.   the stored pattern corresponding to each kernel image is reversely recalled by each separated kernel image, independently,

step4.   Hamming distance is calculated between the input pattern and the reversely recalled patterns obtained in step3,

step5.   the kernel image of the minimum Hamming distance is selected as the plausible kernel image,

step6.   finally, the output pattern is recalled by the final kernel image in the 2nd recall.
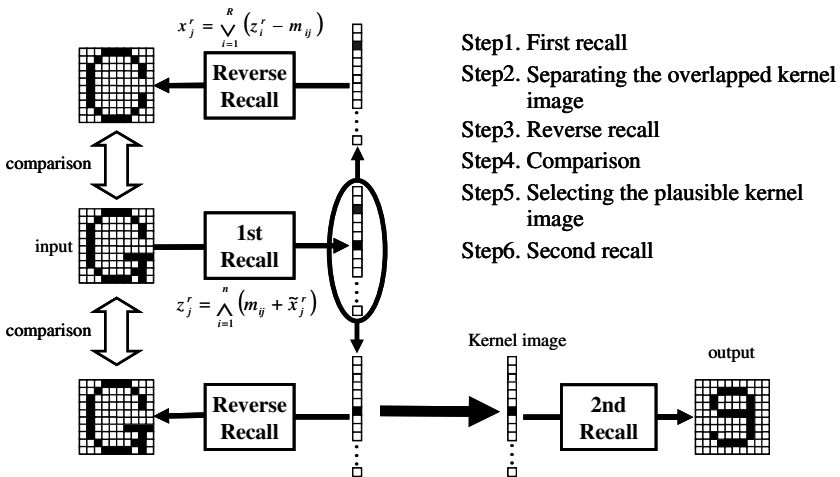
Step1. First recall
Step2. Separating the overlapped kernel image
Step3. Reverse recall
Step4. Comparison
Step5. Selecting the plausible kernel image
Step6. Second recall

$$x_j^r = \bigvee_{i=1}^{R} \left(z_i^r - m_{ij} \right)$$

$$z_j^r = \bigwedge_{i=1}^{n} \left(m_{ij} + \tilde{x}_j^r \right)$$

Fig. 4. Recall process of the MAM employing the reverse recall method [10]

3   Simplified Reverse Recall Method

The MAM employing the reverse recall needs additional calculations for the reverse recall. As the number of included patterns increases, the extra recall time t increases. The extra recall time is given by;

$$t = 0.5 \times N_{ip} \times t_{recall}, \tag{5}$$

where N_{ip} is the number of included patterns and t_{recall} is the recall time with no inclusion pattern. In the MAM employing the reverse recall, the overlapped kernel image is separated to individual kernel images and then the stored pattern corresponding to each kernel image is reversely recalled using each separated kernel image.

Here, in the MAM employing the reverse recall, the stored pattern X^r reversely recalled for the kernel images Z^r is given as;

$$x_j^r = \bigvee_{i=1}^{R} \left(z_i^r - m_{ij} \right), \tag{6}$$

where, $X^r = (x_1^r, \cdots, x_n^r)$, $Z^r = (z_1^r, \cdots, z_R^r)$, x_j^r is j-th unit of the stored pattern X^r, z_i^r is i-th unit of the kernel image Z^r, R is the total number of the stored pattern. In the case of binary patterns, Eq. (6) can be rewritten as;

$$x_j^r = z_r^r - m_{rj}. \tag{7}$$

Eq. (7) shows that the stored pattern corresponding to each kernel image can be reversely recalled by one element of the kernel image. Therefore, by using Eq. (7), the stored pattern can be reversely recalled by the one-shot reverse recall.
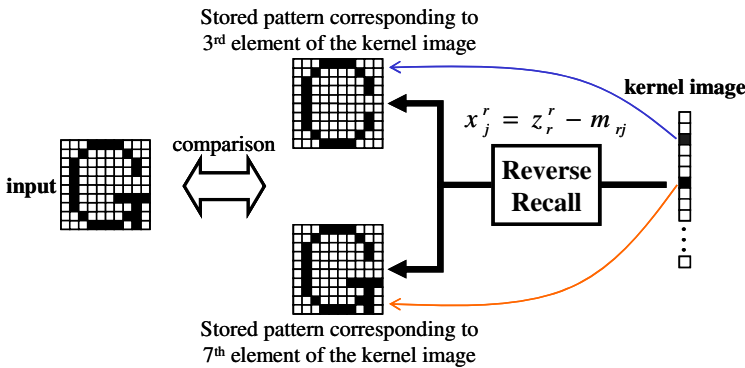
Fig. 5. Reverse recall process in the simplified reverse recall method

Here, the kernel design method of the proposed model uses the stored pattern independent kernel image [9]. In the proposed model, when the overlapped kernel image is recalled in the 1st recall, the stored patterns corresponding to the firing unit of the overlapped kernel image can be reversely recalled by using Eq. (7) because each kernel image is represented by unique one bit. Finally, as shown in Fig. 5, the plausible input pattern is specified by comparing the Hamming distance between the input and recalled patterns and the correct kernel image can be fixed.

The MAM using the proposed method processes through the following steps:

step1. The kernel image is recalled in the 1st recall, as same as the MAM employing the reverse recall,

step2. when the overlapped kernel image is recalled, the stored pattern corresponding to each element of the kernel image is reversely recalled using the corresponding firing unit of the overlapped kernel image, as illustrated in Fig.5,

step3. the final kernel image is determined by comparing an input pattern with patterns obtained in step.2,

step4. finally, the output pattern is recalled using the final kernel image in the 2nd recall.

4   Experimental Result

In order to evaluate the performance of the proposed MAM, we investigate the perfect recall rate and the recall time through hetero-association experiments. In the experiments, each pattern consists of 10 x 10 = 100 binary units. The unit of pattern takes '1' or '0', the '1' represents black and '0' white.

Firstly, we investigate the perfect recall rate of the proposed method for patterns as shown in Fig.6. Fig.6 shows that (a) is the stored patterns that consist of only sets of twofold inclusion patterns and (b) is threefold inclusion patterns. Here, the perfect recall rate is evaluated by an average of 10,000 trials in the simulation. The noise is defined as to change '1' to '0' (or '0' to '1').

(a)   (b)
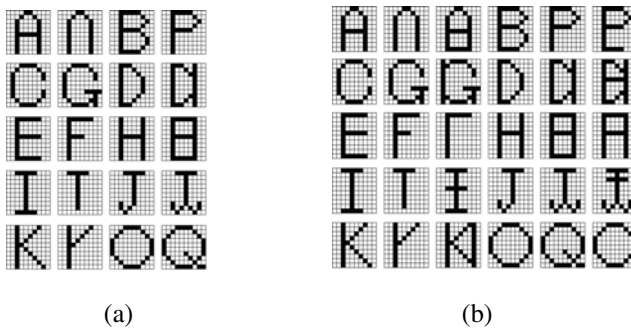
Fig. 6. Stored patterns: (a) patterns that consist of only sets of twofold inclusion patterns, and (b) threefold inclusion patterns

Fig.7 shows the noise tolerance of the MAM using the reverse recall and the proposed method for patterns illustrated in Fig.6 (a). As shown in Fig.7, the performance of the proposed MAM is equivalent to the MAM employing the reverse recall [10].
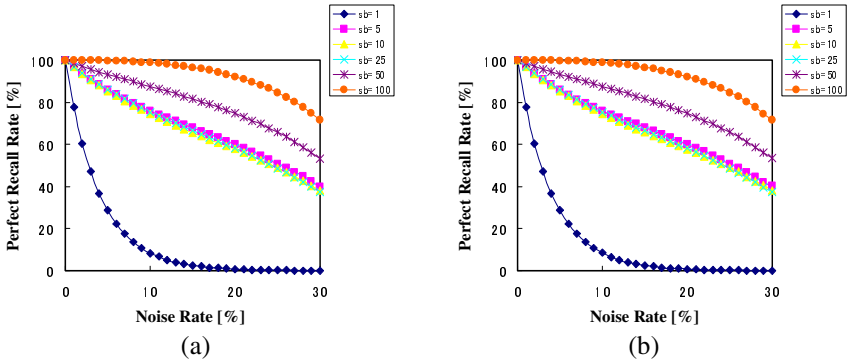
Fig. 7. Noise tolerance in hetero-association for twofold inclusion patterns: (a) the result of the MAM employing the reverse recall, and (b) the proposed method

Next, we investigate the recall time of the MAM using the reverse recall and the proposed method. The one recall time is evaluated by an average of 10,000 trials in the simulation. In the experiment, two sets of patterns illustrated in Fig.6 (a) and (b) are used as the stored patterns. Table1 shows the one recall time of the MAM using the proposed method in comparison with the MAM employing the reverse recall. Here, a CPU of the PC is Intel Core2 Quad @ 2.4 GHz.

Table 1. Recall time of the MAM employing the reverse recall and the proposed method

	One recall time (μ sec)	
	Previous model	Proposed model
Twofold inclusion patterns	83.2	62.4 (25% reduced)
Threefold inclusion patterns	106.5	63.5 (40% reduced)

As shown in Table1, in comparison with the MAM using the reverse recall, the proposed method reduced the one recall time by 25% for twofold inclusion patterns and by 40% for threefold inclusion patterns.

5   Conclusion

The MAM employing the reverse recall has the problem that the extra recall time increases as the number of included patterns increases. In order to overcome the problem, we proposed the MAM employing the simplified reverse recall method. The proposed method reduced the extra recall time without decreasing the noise tolerance. In comparison with the MAM employing the reverse recall, the proposed method reduced the one recall time by 25% for twofold inclusion patterns and by 40% for threefold inclusion patterns. The improved rate increases as the number of included patterns increases.

Although the MAM employing binary stored patterns has been used for many applications, the applications can be extended by using the more complex type of the stored pattern. In the future works, we will develop the model that can handle not only binary patterns but also the other patterns, and tackle practical applications using an associative memory.

References

1. Hopfield, J.J.: Neural network and physical systems with emergent collective computational abilities. Proc. of the National Academy of Sciences of the United States of America 79(8), 2554–2558 (1982)
2. Ackley, D.H., Hinton, G.E., Sejnowski, T.J.: A learning algorithm for Boltzmann machines. Cognitive Science 9, 147–169 (1985)
3. Kosko, B.: Bidirectional Associative Memories. IEEE Trans. Systems, Man and Cybernetics 8(1), 46–60 (1988)
4. Hagiwara, M.: Multidirectional associative memory. In: International Joint Conference on Neural Networks, vol. I, pp. 3–6 (1990)
5. Ritter, G.X., Sussner, P., Diaz-de-Leon, J.L.: Morphological associative memory. IEEE Trans. Neural Networks 9(2), 281–293 (1998)
6. Hattori, M., Fukui, A., Ito, H.: A fast method to decide kernel patterns for Morphological Associative memory. Transactions of the Institute of Electrical Engineers of Japan C 123(10), 1830–1838 (2003)
7. Ida, T., Ueda, S., Kashima, M., Fuchida, T., Murashima, S.: On a method to decide kernel patterns of morphological associative memory, D-II J83-D-II(5), 1372–1380 (2000)
8. Sussner, P.: Associative morphological memories based on variations of the kernel and dual kernel methods. Neural Networks 16(5-6), 625–632 (2003)
9. Harada, H., Saeki, T., Miki, T.: A Morphological Associative Memory Employing A Stored Pattern Independent Kernel Image and Its Hardware Model. In: IWCIA 2009, pp.219–224 (2009)
10. Harada, H., Miki, T.: A Morphological Associative Memory Employing a Reverse Recall. In: Wong, K.W., Mendis, B.S.U., Bouzerdoum, A. (eds.) ICONIP 2010, part II. LNCS, vol. 6444, pp. 321–328. Springer, Heidelberg (2010)
11. Grana, M., Sussner, P., Ritter, G.X.: Innovative Applications of Associative Morphological Memories for Image processing and Pattern Recognition. Mathware and Softcomputing 10(3), 155–168 (2003)
12. Guzman, E., Alvarado, S., Pogrebnyak, O., Yanez, C.: Image Recognition Processor based on Morphological Associative Memories. In: CERMA 2007, pp. 260–265 (2007)
13. Sussner, P.: Generalizing Operations of Binary Autoassociative Morphological Memories Using Fuzzy Set Theory. Journal of Mathematical Imaging and Vision 19, 81–93 (2003)

Analyzing the Dynamics of Emotional Scene Sequence Using Recurrent Neuro-Fuzzy Network

Qing Zhang and Minho Lee

School of Electrical Engineering and Computer Science,
Kyungpook National University,
1370 Sankyuk-Dong, Puk-Gu, Taegu 702-701, South Korea
zhangqing@ee.knu.ac.kr, mholee@knu.ac.kr

Abstract. In this paper, we propose a framework to analyze the temporal dynamics of the emotional stimuli. For this framework, both EEG signal and visual information are of great importance. The fusion of visual information with brain signals allows us to capture the users' emotional state. Thus we adopt previously proposed fuzzy-GIST as emotional feature to summarize the emotional feedback. In order to model the dynamics of the emotional stimuli sequence, we develop a recurrent neuro-fuzzy (RNF) network for modeling the dynamic events of emotional dimensions including valence and arousal. It can incorporate human expertise by IF-THEN fuzzy rule while recurrent connections allow the network fuzzy rules to see its own previous output. The results show that such a framework can interact with human subjects and generate arbitrary emotional sequences after learning the dynamics of an emotional sequence with enough number of samples.

Keywords: Dynamics of Emotion, Electroencephalography (EEG), Fuzzy-GIST, International Affective Picture System (IAPS), Recurrent Neuro-Fuzzy Network (RNF).

1 Introduction

The need for computational and robotic models which can understand the emotional state of the user is ever growing [1]. A lot of literatures are dedicated to studying affect detection. R. Calvo et al give a comprehensive interdisciplinary review of models, methods and applications of affect detection [2]. However, time is clearly important in emotion and emotions are a special dynamic form of cognition, there is few study on analyzing the dynamic events of emotional scene sequence. In this paper, we propose a framework aiming to analyze the temporal dynamics of the emotional stimuli sequence.

In order to make the system be capable of understanding more complex emotions, we consider a valence-arousal (VA) model [3] [4]. By using such a dimensional approach, all emotions can be represented as points in the VA space, in which we can label the images joy, pleasure, anger and sadness. Not only is the VA space helpful in visualizing the location, extent and relationships between emotion categories, but also it is associated with the limbic system which

suggestively supports a variety of functions including emotion, behavior, long term memory, and olfaction. Responses along the emotional valence dimension are associated with significant clusters in the amygdala, the anterior parietal cortex, and the insular cortex. Responses along the arousal dimension are associated with significant clusters in two regions: activity in the right supramarginal gyrus, and thalamic activity varied with reported arousal [5]. For emotion related feature extraction, we adopted the fuzzy-GIST, which is a kind of conceptual gist of a scene that contains semantic information from both EEG and visual information [6]. Furthermore, we develop a novel recurrent neuro-fuzzy network to incorporate the human expertise to model the dynamic events of emotional stimuli sequence.

The remaining sections are organized as follows. Section 2 introduces the proposed fuzzy-GIST and recurrent neuro-fuzzy network for this study. In Section 3, we will give the experiment results and evaluate the performance of the proposed system. Some final conclusions and discussions are given in the last section.

2   Methods

2.1   Overview of the Emotion Dynamic Analysis

Figure 1 demonstrates the graphic outline of the proposed approach. The considered input sequence is split into the sequences of visual information and EEG signal. EEG can be allocated to specific image in the sequence, therefore, both EEG and visual features are dynamic. After signal processing, we extract the fuzzy-GIST for 2-emotion understanding to model the dynamics of emotional valence through a recurrent neuro-fuzzy network. Meanwhile, by taking arousal indicator into consideration, the fuzzy-GIST for 4-emotion understanding are fed to another recurrent neuro-fuzzy network to analyze the dynamic events of the emotional dimension of arousal. The fuzzy-GIST for 2-emotion and that for 4-emotion are introduced in [6]. We can then monitor the emotional trajectory by mapping each stimulus onto VA space.

2.2   Fuzzy-GIST as Emotional Feature at Semantic Level

Since we need features in emotional perspective, we propose the fuzzy-GIST to build a semantic feature vector to represent a scene image as well as consider the human feeling stimulated by the scene. The fuzzy-GIST is originated from the "GIST" [7] [8], and it is a kind of conceptual gist of a scene that contains semantic information. The procedure of extracting fuzzy-GIST from a natural scene is demonstrated in Fig. 2. 11-channel EEG signals are recorded and we adopt wavelet decomposition (WLD) [9] for denoising EEG in our study. The selected wavelet filter for denoising the raw EEG signal is the reverse biothorgonal6.8 (rbio6.8) [10], and we select the D7, D8, and D9 to reconstruct the desired
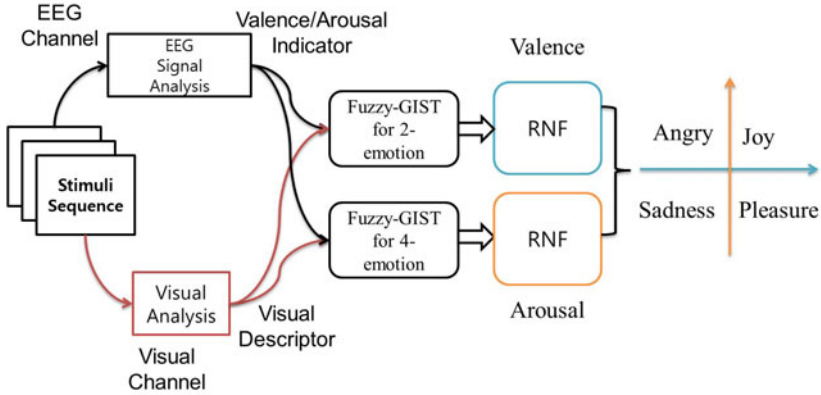
Fig. 1. Graphic outline of the proposed approach

"real signals". We focus on the 500 ms time course starting from stimulus onset and extract the power difference between left and right hemispheres in both alpha and gamma band to monitor the valence state of test subjects [11] [12]. On the other hand, beta/alpha ratio is used as an indicator if the subject is in an arousal state [13]. The EEG features from a subject are processed by the fuzzy C-means clustering (FCM). Based on the clustering result, a natural scene is assigned to positive/negative and calm/arousal groups to a degree of belongingness. According to the relation between the orientation distribution and human emotion evoked by a natural scene, the FCM is used to partition the orientation information of the image into 4 classes in terms of the orientation distribution and make an orientation descriptor for the image. We can describe the lightness as very dark, dark, middle, light and very light. A membership grade maps semantic words because of the fuzziness of human perception [14]. In the similar way, we got warm-cool descriptor including warm, middle and cool, as well as saturation descriptor that indicates the low, middle and high saturation of the natural scene. The brain activity membership grades and visual information membership grades are cascaded to construct the emotional feature space. The difference between the fuzzy-GIST for 2-emotion understanding and that for 4-emotion understanding depends on considering the arousal indicator obtained from EEG signals. Different subjects may have different emotional responses even toward the same scene image, the fuzzy-GIST based on the combination of visual semantic information and the semantic EEG information can help to handle this personal bias for the emotion recognition.

2.3   Recurrent Neuro-fuzzy Network

There is possibility, which is to allow time to be represented by the effect it has on processing. This means giving the processing system dynamic properties
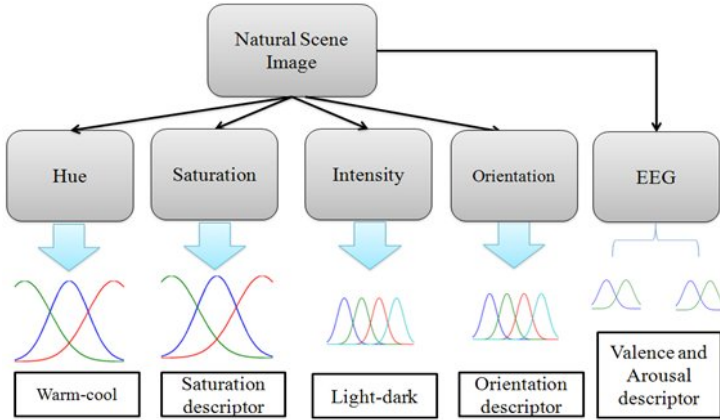
Fig. 2. The procedure of extracting fuzzy-GIST from a natural scene

which are responsive to temporal sequences. In short, for modeling the dynamics of sequence, the network must be given temporal memory.

Figure 3 demonstrates the proposed recurrent neuro-fuzzy network. As show in the Fig. 3, the proposed network is based on Takagi-Sugeno-Kang (TSK) type neuro-fuzzy inference system [14]. The input of the network consists of previous and current input fuzzy-GIST, as well as previous output of the network. Nodes in layer 2 act as membership function to express the input fuzzy linguistic variables, the Gaussian membership function is adopted as:

$$O_{ij}^{(2)} = exp\{-\frac{(u_j^{(2)} - m_{ij})^2}{\sigma_{ij}^2}\} \tag{1}$$

where m_{ij} and σ_{ij} are the center and the width of the Gaussain membership function of the ith term of the jth input variable $u_j^{(2)}$.

Each node in layer 3 is called a rule node, it is formed by fuzzy and operation as:

$$O_i^{(3))} = \prod_{j=1}^{n} O_{ij}^{(2)} = exp\{-\sum_{j=1}^{n} \frac{(u_j^{(2)} - m_{ij})^2}{\sigma_{ij}^2}\} \tag{2}$$

where n is the dimension of input layer.

Nodes in layer 4 are called consequent nodes which perform a weighted linear combination of the input variables. The output of the network is the result of defuzzification (layer 5) of outputs of layer 4. The recurrent connections allow the network's fuzzy rule nodes to see its own previous output, so that the subsequent behavior can be shaped by previous responses. The recurrent connections are what give the networks memory.
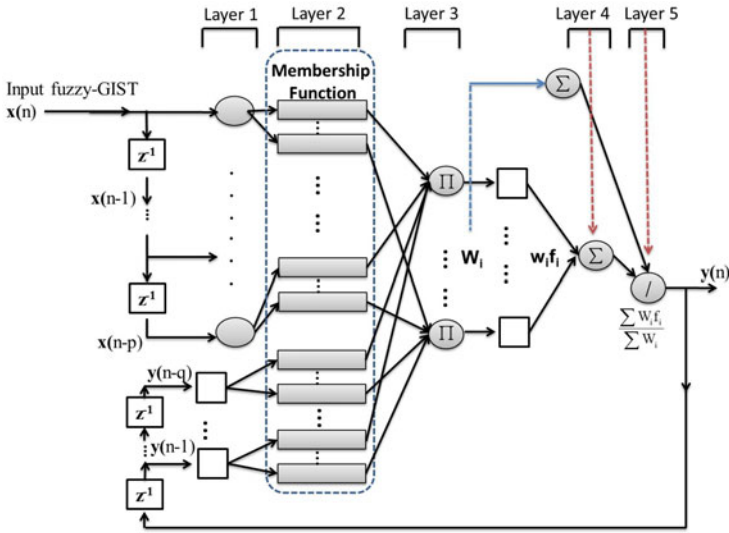
Fig. 3. The architecture of the proposed recurrent neuro-fuzzy network

The input consists of the fuzzy-GISTs of current stimulus and previous 2 stimuli, as well as outputs of previous 5 stimuli. As far as training is concerned, the backpropagation through time (BPTT) algorithm is used [15].

3   Experiment

11 subjects participated in this study and 110 color images selected from IAPS and the corresponding EEG stimulated by images were used to extract the emotional features. We randomly selected 50% of the data and reordered them to generate 20 different emotional sequences for learning the valence and arousal dynamics of the emotional scene sequence. And we used the remaining data to generate random sequence to evaluate the generalization capability of the trained network. During the experiments, human subjects were asked to give valence and arousal scores to describe the emotional feedback of each stimulus. These results are used to evaluate the accuracy of an emotional sequence generated by the network. The current valence and arousal states highly depend on the previous states. In order to handle the temporal effect on emotional processing, we took the average values of the valence and arousal scores of current stimulus and the previous two stimuli as valence value and arousal value for the current sample data. Thus, the valence and arousal scores become more smooth.

Figure 4 shows the learning of dynamic events of an emotional sequence with interacting with a particular subject. The two subfigures on the top show the learning of valence and arousal dynamics, respectively. The solid lines represent

the target responses while the dashed lines represent the dynamic responses of the recurrent neuro-fuzzy networks. On the bottom , the figure on the right-hand side is the emotional trajectory made by locating each stimulus in the valence-arousal space. Compared with the target emotional trajectory shown on the left-hand side, we can see that the networks is capable of learning the dynamics of an emotional sequence.
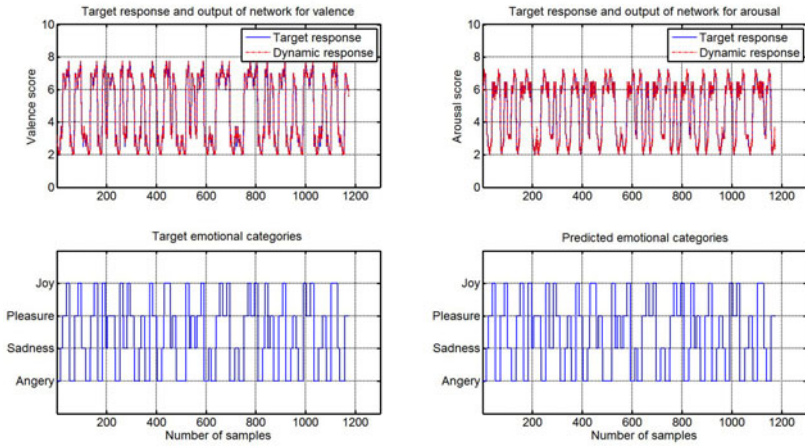
Fig. 4. Learning the dynamics of valence and arousal by two recurrent neuro-fuzzy networks
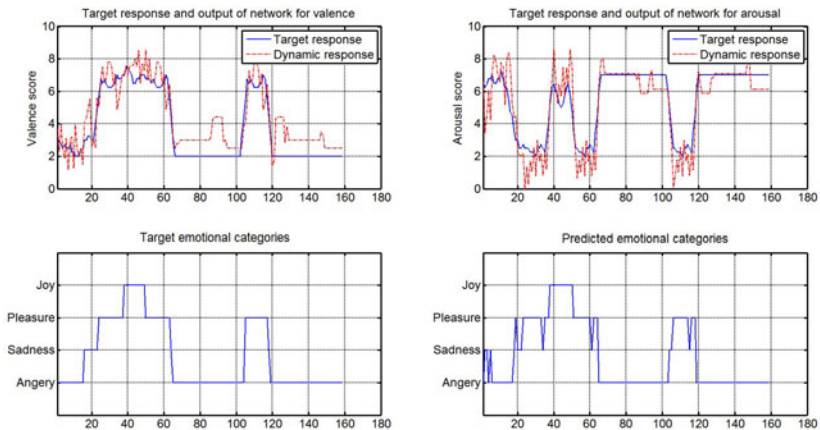
Fig. 5. Valence and arousal targets and output responses for a new sequence and its emotional trajectory
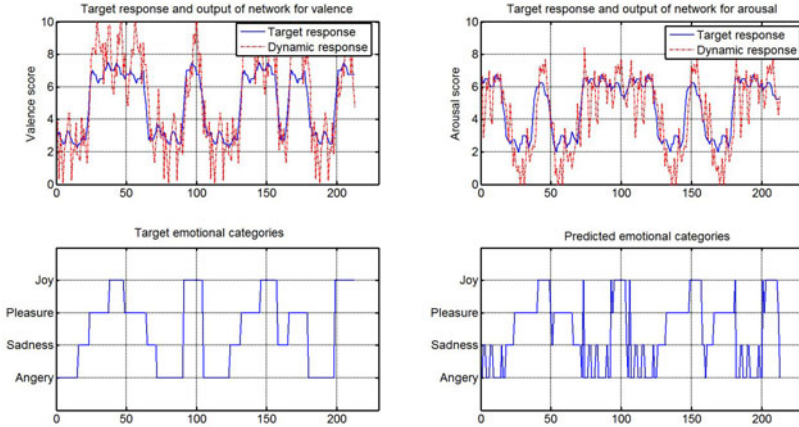
Fig. 6. Valence and arousal targets and output responses for another new sequence and its emotional trajectory

After training the recurrent neuro-fuzzy networks for valence and arousal dimensions, we used unlearnt data samples to randomly generate emotional sequences with different lengths and different orders of emotional categories. Figures 5 and 6 demonstrate the results of the trained recurrent neuro-fuzzy networks generating arbitrary emotional sequences by interacting with the above mentioned subject.

As we can see from Figs. 5 and 6, through modeling the dynamics of valence and arousal dimensions separately, our proposed network is capable of generating the emotional trajectory for a new random emotion sequence.

We repeated the same procedure for each subject for five times, and the average accuracies of generalization for each subject can approximately achieve 70%. This means the framework can interact with human subjects, learn the dynamic events of a sequence of emotional stimuli and then generate a new emotional sequence, which is close to the emotional feedback from human subject.

4   Conclusion and Discussion

A novel framework for analyzing the temporal dynamics of the emotional stimuli is proposed, in which the emotional feature space is built based on the fuzzy-GIST. The proposed recurrent neuro-fuzzy network is capable of separately modeling the dynamic events of emotional valence and arousal and we can then monitor the emotional trajectory by mapping each stimulus onto valence-arousal space. This study shows promising results revealing that machine is capable of interacting with human and generating its own emotion varying with natural scene stimuli.

In the future, we will continue investigating the development of schemes of human emotion dynamics. For the next stage, we will try to consider the dynamic emotional feedback of a subject during watching consecutive natural scene frames or video.

References

1. Picard, R.W.: Toward computers that recognize and respond to user emotion. IBM Syst. J. 39(3-4), 705–719 (2000)
2. Calvo, R., DMello, S.: Affect detection: An interdisciplinary review of models, methods, and their applications. IEEE Transactions on Affective Computing 1(1), 18–37 (2010)
3. Russell, J.A.: A circumplex model of affect. Journal of Personality and Social Psychology 39, 1161–1178 (1980)
4. Lang, P.J.: The emotion probe. studies of motivation and attention. The American Psychologist 50(5), 372–385 (1995)
5. Anders, S., Lotze, M., Erb, M., Grodd, W., Birbaumer, N.: Brain activity underlying emotional valence and arousal: a response-related fMRI study. Human Brain Mapping 23(4), 200–209 (2004)
6. Zhang, Q., Lee, M.: Emotion development system by interacting with human eeg and natural scene understanding. Cognitive Systems Research (in Press, Corrected Proof, 2011)
7. Oliva, A., Torralba, A.: Building the gist of a scene: the role of global image features in recognition. Progress in Brain Research 155, 22–36 (2006)
8. Zhang, Q., Lee, M.: Analysis of positive and negative emotions in natural scene using brain activity and gist. Neurocomputing 72(4-6), 1302–1306 (2009)
9. Burrus, S.C., Burrus, S.C., Gopinath, R.A.: Introduction toWavelets andWavelets Transforms. Prentice-Hall (1997)
10. Mallat, S.G.: A theory for multiresolution signal decomposition: the wavelet representation. IEEE Transactions on Pattern Analysis and Machine Intelligence 11, 674–693 (1989)
11. Niemic, C.P.: A theoretical and empirical review of psychophysiological studies of emotion. Journal of Undergraduate Research 1, 15–18 (2002)
12. Müller, M.M., Keil, A., Gruber, T., Elbert, T.: Processing of affective pictures modulates right-hemispheric gamma band eeg activity. Clinical Neurophysiology 110(11), 1913–1920 (1999)
13. Kandel, E.R., Schwartz, J.H., Jessell, T.M.: Principles of Neural Science. McGraw-Hill Medical (2000)
14. Jang, J.S.R., Sun, C.T., Mizutani, E.: Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence. Prentice Hall, Inc., Upper Saddle River (1997)
15. Haykin, S.: Neural Networks: A Comprehensive Foundation, 2nd edn. Prentice-Hall (1998)

Stress Classification for Gender Bias in Reading

Nandita Sharma and Tom Gedeon

Research School of Computer Science,
Australian National University,
Canberra, Australia
{Nandita.Sharma,Tom.Gedeon}@anu.edu.au

Abstract. The paper investigates classification of stress in reading for males and females based on an artificial neural network model (ANN). An experiment was conducted, with stressful and non-stressful reading material as stimuli, to obtain galvanic skin response (GSR) signals, a good indicator of stress. GSR signals formed the input of the ANN with stressed and non-stressed states as the two output classes. Results show that stress in reading for males compared to females are significantly different ($p < 0.01$), with males showing different patterns in GSR signals to females.

Keywords: classification, artificial neural networks, galvanic skin response, gender stress, reading.

1   Introduction

The term, *stress*, was first coined by Hans Selye, who defined stress as "the non-specific response of the body to any demand for change" [1]. Stress is defined as the body's psychological and physical reaction or response to the imbalance between demands and the resources available for a person. It is known as a natural alarm, resistance and exhaustion [2] system for the body to prepare for a fight or flight response to protect the body in case of threats and make the body adapt to changes. Stress has been identified as a serious and growing issue adversely impacting both individuals and society. It is widely accepted that stress has the potential to cause a variety of chronic illnesses ranging from cardiovascular diseases, diabetes and even some forms of cancer. Societies are affected by the stress problem due to large proportions of their citizens facing stress, which is resulting in high economic costs, especially in developed countries [3, 4]. Stress research has a wide range of potential applications including the capacity to improve personal operations, learning and increase work productivity [5], making it an interesting area of research in a variety of fields including posing a technical challenge to Computer Science. A number of computational techniques have been applied to define stress objectively and build simplistic stress models based on techniques such as Bayesian networks [6], support vector machines [7], and decision trees [8].

   Due to complexity in its definition, stress cannot be measured directly and objectively but can be modeled in terms of other measures, including galvanic skin response (GSR) [9], or heart rate variability [10]. GSR, also known as *skin*

conductance or *electrodermal activity response*, is reliable as a measure of stress [10-12] and it is used in this paper to classify stress. When an individual is under stress, skin conductance is increased [6], which may result in moisture on the surface of the skin, or in a more extreme case leading to sweating. The increase in skin conductance is recorded as a change in GSR. GSR has been used as a measure in a biofeedback system to help individuals lower their stress levels. The system had Biopac sensors to take GSR readings and a competitive racing game component that induced stress [13].

Artificial Neural Networks (ANN) has been used for modeling stress but is at early stages of research. Physiological signals have been used as inputs to ANN for classifying mental workload with good accuracy rates [14]. Some emotions, such as fear and anger, are symptoms of stress, which have been differentiated using ANNs with GSR as inputs. This paper presents an ANN for determining whether there is a gender based difference in stress for reading. It uses GSR data obtained from a reading experiment comprising of *stressed* and *non-stressed* reading to obtain data sets for male, female and a combined data set, develops an ANN classifier for the data sets, and then provides an analysis of the results. An experiment was conducted, with stressful and non-stressful reading material as stimuli, to obtain galvanic skin response (GSR) signals, which is a good indicator of stress. GSR signals formed the input of the ANN with stressed and non-stressed states as the two classes. Three ANNs were built to investigate whether there is gender bias in stress. Results from the experiments show that stress in reading for males and females are significantly different ($p < 0.01$). The paper evaluates the classifiers, validates them, analyses classification rates, concludes the findings, and proposes further research.

2   Reading Experiment

Undergraduate Computer Science students over the age of 18 years old were recruited as experiment participants (after obtaining Ethics Approval from the Australian National University Ethics Committee). The participant group for the investigating gender bias experiment was made up of 10 males and 10 females with age between 18 and 24 years. Each participant had to understand the requirements of the experiment from written experiment instructions with the guidance of the experiment instructor before they filled in the experiment consent form. Afterwards, GSR electrodes were attached to the participant. The instructor notified the participant to start reading, which triggered a sequence of text paragraphs. After finishing the reading, participants had to do an assessment. An outline of the process of the experiment for an experiment participant is shown in Fig. 1.

Fig. 1. Overview of the experiment process

2.1 Experiment Set Up

Experiment participants had to silently read *stressed* and *non-stressed* types of paragraphs. Stressed paragraphs had stressful content, whereas the non-stressed paragraphs had content that created an illusion of meditation or soothing environments. There were three paragraphs for each type of paragraph and each paragraph was about the same length, approximately 360 words. Participants were told that they would be monitored on the way they interacted with text while they read and then had to do an assessment, based on what they read, after the reading.

The paragraphs were displayed on a 1050 x 1680 pixel Dell monitor. For consistency, paragraphs were displayed on the screen for 60 seconds and positioned at the same location of the computer screen for each participant. Each line of the paragraph had 70 characters.

Biopac GSR100C equipment was used to obtain GSR signals at 1 kHz. Disposable EL507 GSR electrodes were used, which were placed on the first (or index) and forth (ring) fingers of the participant's left hand.

Fig. 2. The experiment was conducted in a room with a consistent environment for participants. A computer monitor displayed the reading text. The participants wore GSR equipment on their left hand, which they rested on the arm rest while reading.

3 ANN Classifier for Determining Gender Bias

An ANN is inspired by biological neural networks with characteristics for learning and reacting. It is made up of interconnected processors, known as *artificial neurons*, connected by weighted links that pass signals between neurons. Feed-forward ANNs were developed, trained and tested for testing the null and alternative hypotheses:

H1. Null: There is no difference in GSR patterns for females and males during reading stressed and non-stressed material.

H2. Alternative: There is a difference in GSR patterns for females and males during reading stressed and non-stressed material.

A three-layer (plus the input layer) ANN was trained on a data set obtained from the experiment to test the hypotheses. Each of the ANNs developed was trained on one of the three data sets:

DF. Female data set – standardised GSR data for all female participants

DM. Male data set – similar to the female data set, but the data set contained data for male participants only

DFM. Male and Female data set – composed of the male and female standardized GSR data sets

An ANN was developed by training it on the male and female data set (DFM) to test H1. Classification rates for participants were obtained by swapping two females and males at a time from the training and testing sets at a time. In order to test H2, two ANNs were developed. Each ANN was trained on either the female data set (DF) or the male data set (DM). Classification rates of a participant was determined on the ANN trained using the opposite gender e.g. female participants were classified using the ANN that was trained on DM.

The Sigmoid function was chosen as the activation function to limit the output values of the ANN within the range of 0 (not-stressed) and 1 (stressed). The ANNs were developed using the Matlab Neural Networks Toolbox and trained using the Levenberg-Marquardt back-propagation algorithm.

In order to minimize individual bias, the GSR signals were standardized to a range within 0 and 1 using Equation (1) for each timestamp T(i).

$$Standardised\_GSR_{T(i)} = (Raw\_GSR_{T(i)} - Minimum\_GSR_{T(i)})/RangeGSR_{T(i)} \tag{1}$$

The size of the raw GSR data set for a participant reading a particular type of paragraph was 180,000, which was a result of a sampling rate of 1 kHz. This data set is not feasible to extract suitable features. Therefore, a maximum value was obtained for every 1000th interval to reduce the data set size to 180. This gave a data set a granularity level of one second. Subsequently, a running maximum algorithm was used with a window of five seconds. GSR after each second, $SGSR_t$, was defined as the maximum $Standardised\_GSR_{T(i)}$ from a window of size five as described in Equation (2).

$$SGSR_t = \max\{Standardised\_GSR_{T(j)}\} \text{ where } i <= j < i+5 \tag{2}$$

Each $SGSR_t$ formed an input to the ANNs. Research has showed that physiological responses affected by stress appears a short time after an individual has been exposed to stressful stimulus [10, 15]. The experiment data, depicted in Fig. 4, shows an example of the claim. Also, a participant was given 60 seconds to read a paragraph, so as a consequence, $SGSR_t$ values over a 30 second time interval was used to develop a sample in the data set for the ANNs. An input list, I_k, for a sample in the ANN data set is defined in Equation (3).

$$I_k = [SGSR_k, SGSR_{k+1}, SGSR_{k+2}, \ldots, SGSR_{k+29}] \tag{3}$$

Classification of stress for reading is non-linear and the input fed to the ANN was discretised and segmented. The topology of the ANN was obtained empirically based

on capturing prominent characteristics appearing in the data (described in Section 4) and experimental trials. The characteristics were observed over approximately 30 second time intervals so GSR data over 30 second time intervals formed inputs to the ANN, thus the ANN had 30 input neurons. Rules of thumb were incorporated in determining the number of neurons in the hidden layers of the ANN. The first hidden layer was designed so that the number of neurons was no more than twice the number of neurons in the input layer and the number of neurons in the second layer was less than the square root of the number of neurons in the first hidden layer. Accordingly, 50 neurons were allocated to the first hidden layer, 7 neurons were in the second hidden layer and one neuron in the final layer. Moreover, the topology was defined so that the ratio of free parameters to the number of training patterns was at most 3.

4   Results and Discussion

The GSR data for reading stressed and non-stressed material obtained from the reading experiment for each participant is shown in Fig. 3. By observation, prominent characteristics of stress, spikes or short periods with high GSR values, in the data
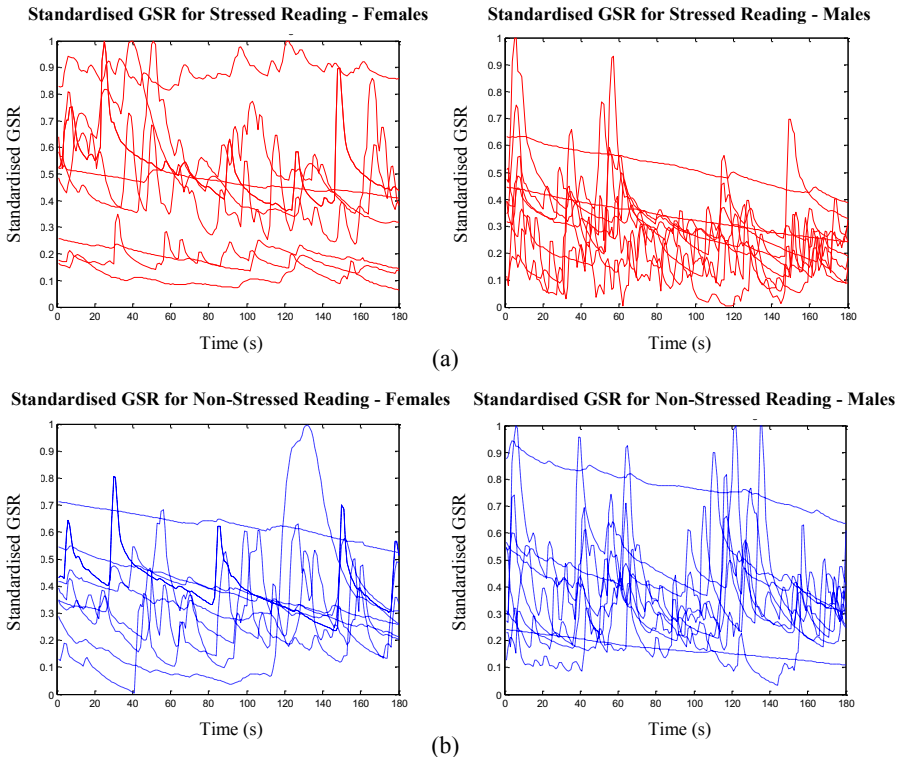
Fig. 3. Standardised GSR values were obtained for participants in the reading experiment. Each participant read three stressed and three non-stressed paragraphs in consecutive order for stressed and non-stressed readings respectively. (a) Standardised GSR values for male and female data sets while stressed reading (b) Standardised GSR values for male and female data sets while non-stressed reading.

generally span more than 20 seconds. Accordingly, the inputs for the ANN were defined to include this span, but not a span as large as the time period given to read a paragraph, which was 30 seconds. This generated 6,040 data samples, which were used to train the ANN.

A 10-fold cross-validation method was used to evaluate the ANNs trained on DF, DM and DFM. The ANN that was trained on the female data set performed the best. The ANN that was trained on the male data set did not perform much worse, however, the ANN which was trained on the training set with both the male and female data performed the worst. This variation suggests that characteristics in male and female data sets may not be similar. The performance values of the ANN on the three different training sets are compared in Table 1.

Table 1. Statistics for 10-fold cross-validation on the ANN for the reading data sets

Data set	Correct rate	Sensitivity	Specificity
Female	91%	91%	91%
Male	81%	77%	84%
Male and Female	70%	71%	70%

Average classification rates for correct classification for each of the participant samples were calculated over a set of five trials. For each trial, the ANN was trained using the training set, which varied for the hypotheses that were tested. The classification rates for stress for male and female participants on the ANN trained on DFM is shown in Table 2.

Table 2. Average correct classification rates for samples tested on the ANN trained using samples of both the genders

Participant ID	Correct classification rates for female samples	Correct classification rates for male samples
1	68%	50%
2	33%	34%
3	42%	62%
4	71%	60%
5	33%	64%
6	55%	58%
7	76%	56%
8	51%	15%
9	78%	14%
10	31%	44%

The Student's t-Test gives a p-value of 0.004 ($p < 0.01$), which shows that we have strong evidence that the null hypothesis, H1, does not hold. As a result, there is no evidence that suggests that there are no differences in GSR patterns for females and males during reading stressed and non-stressed material, which suggests investigating existence of difference in GSR patterns.

The classification rates for stress for each female participant on the ANN trained on DM and for each male participant on the ANN trained on DF are given in Table 3 to determine whether there is a difference in patterns in GSR for males and females.

Table 3. Average correct classification rates for samples tested on the ANN trained using samples of the opposite gender

Participant ID	Correct classification rates for female samples	Correct classification rates for male samples
1	56%	54%
2	62%	24%
3	34%	45%
4	45%	54%
5	27%	43%
6	43%	54%
7	38%	60%
8	8%	10%
9	47%	34%
10	38%	35%

The classification rates for male and female samples have a p-value of 0.16 ($p > 0.1$), which means that there is strong evidence that males show different characteristics and patterns in GSR signals for stress to females. Thus, H2 hypothesis holds. Classification results in Table 2 and Table 3 and statistical analyses show that males and females respond to stress in reading with different GSR signals.

5   Conclusion and Future Work

ANN was used to classify stress in reading for females and male. We showed that stress in males and females are significantly different ($p < 0.01$) i.e. gender has influence in the separability of GSR characteristics. The results are consistent with previous studies [16]. Future possible extension to the reported work includes modeling GSR features for males and females on other types of classifiers (e.g. SVM) to generalize reported patterns, investigating discrepancies within the male data set and developing a method for obtaining an improved classifier for a male stress data set. In order to obtain a classifier with a higher classification rate, inputs of an ANN could be defined in terms of multiple primary measures of stress e.g. a combination of GSR and HRV measures.

References

1. Selye, H.: Confusion and controversy in the stress field. Journal of Human Stress 1(2), 37–44 (1975)
2. Hoffman-Goetz, L., Pedersen, B.K.: Exercise and the immune system: a model of the stress response? Immunology Today 15(8), 382–387 (1994)
3. The-American-Institute-of-Stress. America's No. 1 Health Problem - Why is there more stress today? (August 5, 2010), http://www.stress.org/americas.htm
4. Lifeline-Australia, Stress Costs Taxpayer $300K Every Day (2009)
5. Kompier, M., Cooper, C.L.: Preventing stress, improving productivity: European case studies in the workplace. Routledge (1999)

6. Liao, W., et al.: A real-time human stress monitoring system using dynamic bayesian network (2005)
7. Dou, Q.: An SVM ranking approach to stress assignment. University of Alberta (2009)
8. Zhai, J., Barreto, A.: Stress recognition using non-invasive technology (2006)
9. Shi, Y., et al.: Galvanic skin response (GSR) as an index of cognitive load. ACM (2007)
10. Healey, J.A., Picard, R.W.: Detecting stress during real-world driving tasks using physiological sensors. IEEE Transactions on Intelligent Transportation Systems 6(2), 156–166 (2005)
11. Labbé, E., et al.: Coping with stress: the effectiveness of different types of music. Applied Psychophysiology and Biofeedback 32(3), 163–168 (2007)
12. Ferreira, P., et al.: License to chill!: how to empower users to cope with stress. ACM (2008)
13. Bersak, D., et al.: Intelligent biofeedback using an immersive competitive environment. In: 2001: Paper at the Designing Ubiquitous Computing Games Workshop at UbiComp. (2001)
14. Wilson, G.F., Russell, C.A.: Real-time assessment of mental workload using psycho physiological measures and artificial neural networks. Human Factors: The Journal of the Human Factors and Ergonomics Society 45(4), 635 (2003)
15. Rimmele, U., et al.: Trained men show lower cortisol, heart rate and psychological responses to psychosocial stress compared with untrained men. Psychoneuro endocrinology 32(6), 627–635 (2007)
16. Wang, J., et al.: Gender difference in neural response to psychological stress. Social Cognitive and Affective Neuroscience 2(3), 227 (2007)

Self-Adjusting Feature Maps Network

Chin-Teng Lin, Dong-Lin Li, and Jyh-Yeong Chang

Institute of Electrical and Control Eng.,
Natinal Chiao Tung University,
Hsinchu, Taiwan
{ctlin,jychang}@mail.nctu.edu.tw,
lazybones000@yahoo.com.tw

Abstract. In this paper, we propose a novel artificial neural network, called self-adjusting feature map (SAM), and its unsupervised learning algorithm with self-adjusting mechanism. After the training of SAM network, we will obtain a map composed of a set of representative connected neurons. The trained network structure of representative connected neurons not only displays the spatial relation of the input data distribution but also quantizes the data well. SAM can automatically isolate a set of connected neurons, in which the number of the set may indicate the number of clusters to be used. The idea of self-adjusting mechanism is based on combining of mathematical statistics and neurological advance and retreat of waste. For each representative neuron, there are three periods, growth, adaptation and decline, in its training process. The network of representative neurons will first create the necessary neurons according to the local density of the input data in the growth period. Then it will adjust neighborhood neuron pair's connected/disconnected topology constantly according to the statistics of input feature data in the adaptation period. Lastly the unnecessary neurons of the network will be merged or deleted in the decline period. In this study, we exploit SAM to handle some peculiar cases that cannot be well dealt with by classical unsupervised learning networks such as self-organizing feature map (SOM) network. Furthermore, we also take several real world cases to exhibit the remarkable characteristics of SAM.

Keywords: Unsupervised learning, self-adjusting, representative neurons statistics, SOM.

1 Introduction

In machine learning, unsupervised learning is a class of problems in which one seeks to determine how the data are organized. With some network architectures it is possible to map patterns of arbitrary dimensionality (the pattern space) onto a lower-dimensional structure of neurons (the feature space) having similar topology relations. A well-known approach to designing a self-organizing feature map (SOM) network is to use the Kohonen learning rule [1]. The Kohonen's SOM is giving a topology-preserving map that preserves neighborhood relations of the input pattern to ask for nearby outputs of a feature map corresponding to nearby input patterns, and it updated weights going to the neighbors of the winning neuron as well as to the winning

neuron itself. However there are two critical issues about SOM. First, the network becomes unstable when input patterns have a complex structure. Second, it may easily happen in SOM that the weight vectors lying in zero-density areas are affected by input vectors from all the surrounding parts of the non-zero distribution. To remove such problems [2], the Parallel Distributed Processing (PDP) model [3] and fuzzy self-organizing neural network [4] were proposed. To further deal with stability plasticity dilemma problem, some self-creating models were proposed [5-11]. Besides, we cannot ensure the similarity of network structures after the training process of the same input patterns are in random inputting order. The self-organizing motor map by Ritter et al. [12], and its enhanced version of the Clusot algorithm [13] can utilize the information contained in a trained SOM and do automatic detection of clusters in this surface. Furthermore, the visualization of most networks was proposed cannot directly represent the data structure and inter-neuron distance in its network topology preserved mapping. Many paper are proposed visualize SOM for various concepts [14-17].

In this paper we proposed a self-organizing and self-varying feature map network with a novel learning rule. To obtain a more adaptive neural network, the learning rule combined the topology-preserving map with the self-adjusting mechanism. The self-adjusting mechanism is based on statistics that evaluate what size and structure should the network contain without giving a lot extra parameters. The neural network with the learning rule we proposed can dynamically change the size of neural network and constantly adjust the neighborhood relations of neurons according to input feature data in the training process. Therefore, it can ensure the networks of different trainings will be similar when the training data is the same. Besides, the visualization on trained result can obviously show distribution relations in the input feature space by the neighborhood relations among output neurons. In other words, we can automatically evaluate how many classes the data samples are without any post process.

The remainder of this paper is organized as follows. In Section 2, we describe how to dynamically change the size of neural network and adjust the neighborhood relations of neurons in training process by the self-adjusting mechanism. Furthermore we also introduce the learning rule and the parameters used in our network. In Section 3, we will discuss the effect on different parameters. Section 4, we show the experimental results of our neural network. Finally, in Section 5, we give our conclusions and we discuss some future work.

2   Self-Adjusting Mechanism

In our neural network, we established a self-adjusting mechanism to dynamically adjust network size and structure according to the input data distribution. There are two aspects in self-adjusting mechanism: The adaptive network size and the dynamic neighborhood topology.

Assume input vector is $\bar{x}_i(t)$ for $i = 1, \cdots, M$ and the weight vector of representative neuron j is $\bar{w}_j(t) = \left[\bar{w}_{j1}, \bar{w}_{j2}, \cdots, \bar{w}_{jq} \right]^T$ for $j = 1, \cdots, N$, where M and N represent the total number of training data and the representative neuron number

respectively. Parameters q and t represent the dimension of the input vector and the learning time elapsed, respectively. Since the learning time scale is proceeded with a epock-based approach, thus the learning time t elapsed can be written as $t = nT_{ep}$, in which T_{ep} is the training time needed for an epock learning of all input data learned once. Hereafter, we will equivalently write $t = nT_{ep} \doteq n$ (discrete domain) for brevity. The input vector is compared with all the weight vectors. A best-matching unit (BMU) $C_i(t)$ can be found by calculating the Euclidean distance between the input vector and the weight vector $\bar{w}_j(t)$ of representative neuron j, i.e.,

$$C_i(t) = \arg \min_j \{ \| \bar{x}_i(t) - \bar{w}_j(t) \| \} \tag{1}$$

After the training iteration, the standard deviation $STD_j(t)$ of data subset whose winner is neuron j is define as

$$STD_j(t) = \frac{\sqrt{\sum_{i=C_i \in \text{Neuron}_j} \left(\bar{x}_i - \bar{w}_j(t) \right)^2}}{m} \tag{2}$$

where m and t, respectively, represent the number of the input belong to neuron j and current iteration learning time.

2.1   Learning Rule

In common with Kohonen's SOFM, we give a fixed gird topology of neurons at first. At each iteration, input vectors are drawn randomly and presented to the network. In order to fully utilize self-adjusting mechanism to adjust the network size and structure through the learning of the input data automatically. Each representative neuron j of the SAM network assumes a learning rate function $\alpha_j(t_j)$ and neighborhood updating region $R_j(t_j)$ The learning rate function neuron j might have been more persuasive if it follows an exponential decay function as

$$\alpha_j(t_j) = \exp\left(-\lambda\, t_j \right) \tag{3}$$

where $\lambda \in (0,1)$ is the parameter for controlling representative neuron j activity length and t_j is neuron j learning time being elapsed since its creation, i. e., equivalently the learning epoch n_j being proceeded. When all the representative neuron's learning rate $\alpha_j(t_j)$ approaches zero, the training has completed. The SAM network, including its size and structure, has reached its stable steady state through the learning of the input data. To ensure that the network will be stable and convergent, we divide self-adjusting mechanism of each representative neuron into three periods according to its respective learning rate function as shown in Fig. 1. When the learning rate of a neuron descends from 1 to 0.5, the neuron is called in the *growth* period. A new neuron will be created if the parent representative neuron is not centralized enough. When the learning rate falls between 0.5 and 0.1, it is in the *adaptation* period and the neuron start to evaluate its neighborhood relation, either

connected or disconnected should be established for its neighbor. At last, when the learning rate becomes smaller than 0.1, the neuron is in the *decline* period and the unnecessary neurons will be deleted or merged. The network will terminate learning when the learning rate of each neuron approaches zero, setting to 0.01 in this paper. Note that λ is usually smaller than 0.1. For example, if λ equals 0.08, the representative neuron j is in the *growth* period when $0 \leq n_j \leq 8$; the neuron is in the *adaptation* period when $8 < n_j \leq 28$; the neuron is in the *decline* period when $29 < n_j \leq 52$.

Moreover, as SOM, the neighborhood updating region function is

$$R_j(n_j) = R_{init} \times \alpha_j(n_j) \tag{4}$$

where R_{init} is the initial neighborhood updating region. After the BMU is determined, the weight vectors of the neurons are modified according to the update rule formulated as

$$\bar{w}_j(n_j+1) = \begin{cases} \bar{w}_j(n_j) + \alpha_j(n_j)h_{cj}(n_j)[\bar{x}(t) - \bar{w}_j(n_j)], & j \in R_c(n_j) \\ \bar{w}_j(n_j), & \text{otherwise} \end{cases} \tag{5}$$

where $h_{cj}(T)$ is the neighborhood function and it is centered on the best-match neuron $\bar{w}_c$ and decreasing monotonically with time. A typical smooth neighborhood function could be a Gaussian function:

$$h_{cj}(n_j) = \exp\left[\frac{-\|\bar{x} - \bar{w}_c\|^2}{2\sigma(n_j)^2}\right] \tag{6}$$

where $\sigma(T)$ is the width of the Gaussian kernel and $\|\bar{x} - \bar{w}_c\|^2$ is the distance between the winning neuron $\bar{w}_c$ and the input data $\bar{x}$ on the topology of neighborhood. The learning process consists of winner selection by (1) and adaptation of the weight vectors by (5).
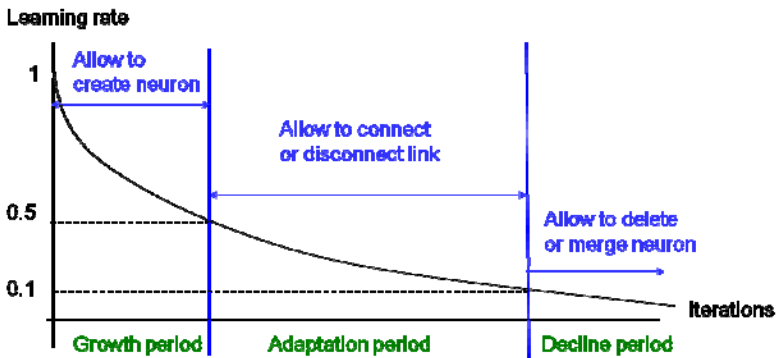
Fig. 1. Stages of the self-adjusting mechanism of a representative neuron

2.2   Adaptive Network Size and Structure

There are three periods for representative neurons which lead to adaptive network size and structure of SAM dynamically. The three periods for representative neurons include neuron growing, neuron adaptation, and neuron decline, which are described below.

Neuron Growing Period. In the neuron *growing* period, wherever the representative neuron is not centralized enough, an new representative neuron will be created. We count the data subset S_j belonging to representative neuron j. If the data subset S_j that fall within d (usually set as 1.5 to 2) times standard deviations from the representative neuron j is smaller than p percentage, such as 80%, as shown in Fig. 2, we create a new neuron form neuron j, which is called the representative neuron k here. This representative neuron can be chosen randomly from the data subset S_j that also falls outside d (usually set as 1.5 to 2) times standard deviation.

The representative neuron k will start its local learning curse as shown in Fig, 1 with its local time setting as $t_k = 0$. By this way, the SAM network with a reasonable parameter setting always reaches the adaptive size and structure that was determined automatically by input data distribution. However, in some classical applications such as vector quantization for image coding, the codebook size, i.e., the network size can only be pre-specified.
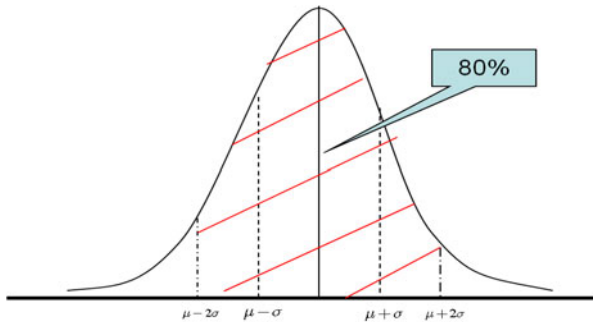
Fig. 2. The condition of neuron growing

Neuron Adaptation Period. In *adaptation* period of the representative neighboring neurons, there are two states for dynamic neighborhood topology of connectivity: disconnected state indicating neighborhood splitting and connected state indicating neighborhood joining. Fig. 3 shows the conditions of neighborhood splitting and joining. If the Euclidean distance between two neighboring neurons is greater than the sum of the two neurons' STDs multiplied by a given value k_{con}, we will disconnect the linking between the neurons for they are not close enough. In other words, they are not close enough neighbor. Otherwise, we will connect the linking between these two neurons for they are close enough. The equation to define connectivity of neighboring neurons is given by

$$\left\| \bar{w}_{il}(t) - \bar{w}_{jl}(t) \right\| \le k_{con} \times \left(STD_{il}(t) + STD_{jl}(t) \right), \quad l = 1, \ldots, q. \tag{7}$$

If the above condition is satisfied, connect the link between the neurons. Otherwise, disconnect them.
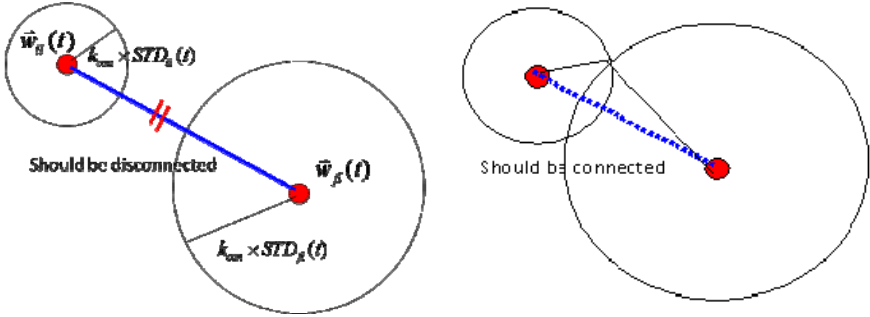
(a) The condition of neuron pairs to become disconnected

(b) The condition of neuron pairs to become connected

Fig. 3. The conditions of neighboring neurons to become connected or disconnected

Neuron Decline Period. In the neurons *decline* period, if the Euclidean distance between two neurons is smaller than the larger STD of two neurons that multiplied by a preset value k_{merge} (usually greater than 2), to merge two neurons into one neuron with the neighborhood relations completely inherited from neurons as shown in Fig. 4. The equation to define the merging of neighboring neurons is given by

$$\left\| \bar{w}_{il}(t) - \bar{w}_{jl}(t) \right\| \le k_{merge} \times \max \left(STD_{il}(t), STD_{jl}(t) \right), \quad l = 1, \ldots, q. \tag{8}$$

If the above condition is satisfied, we merge the neuron i and neuron j.

For neuron deleting case, if the number of the input belong to neuron j is smaller than a threshold β, setting to 0.05 in this paper, then we delete the neuron j.
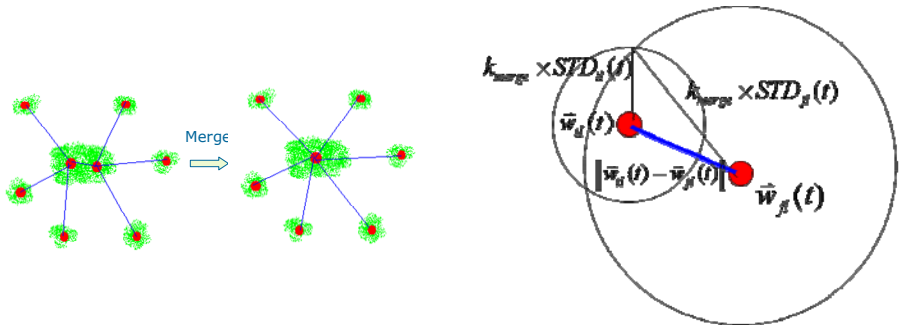
(a) The plot for merging operation

(b) The condition of neuron merging

Fig. 4. The conditions of neurons merging

3   Experiments

In this section, we show the experimental result of simulation and the real world application. In the first part, we conduct experiment on synthetic data set and in the second part, we use a real world application.

Handling Nonstationary Data Input. In the first simulation, Data set 2 contains two clusters of Gaussian data with $\sigma = \sqrt{10}$ and $\sigma = \sqrt{20}$ respectively, and both cluster have 300 samples. The parameters $(p, d, k_{merge}, k_{con})$ of SAM are set as (0.9, 2, 2.2, 2.5); The neighborhood topology of SOM is 4×4 grid; The parameter of BCL is set as $d_0 = 0.3$. The blue lines between neurons indicate that they are due to topological neighborhood relation and the yellow indicates the unit STD range of data subset governing by winning neurons. Figs. 5(b)–(d), respectively, show the training result of SAM, SOM, BCL with 16 neurons for date set 1. As usual, we use the mean square error (MSE) criterion to evaluate performance of each result. Experimental results show the MSE of SAM is smaller than other networks. Fig. 5(e) shows the average MSEs in ten consecutive experiments on the data set 1, when the network size is set as 4, 16, 64, 128, 256 neurons, respectively. We can see form Fig. 5(e) that the MSE of SAM is better than others independently of what the network size is.

(a)   (b)   (c)

(d)   (e)
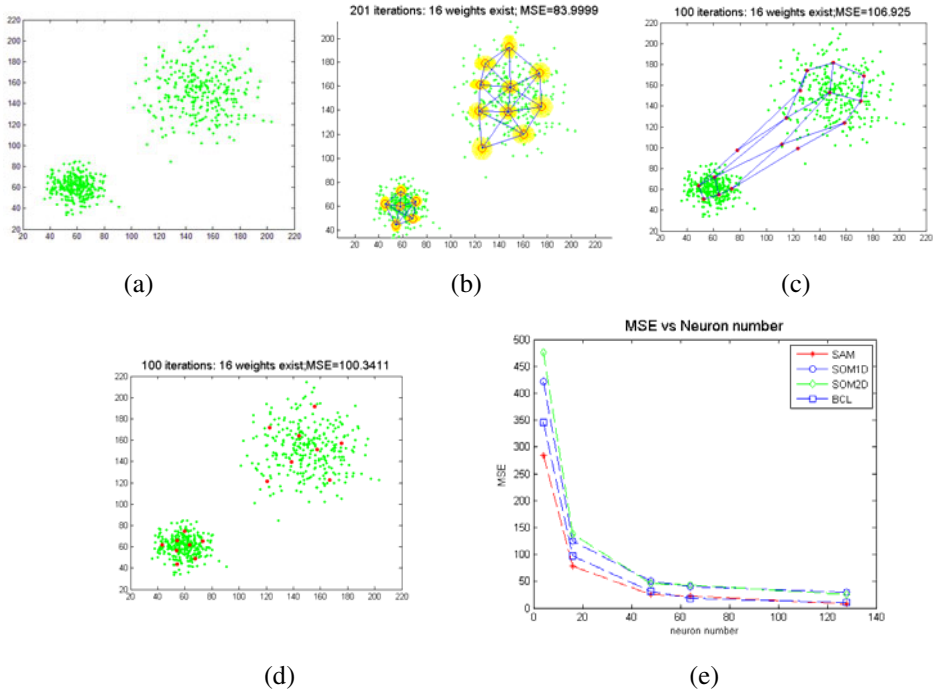
Fig. 5. Nonstationary data input and the vector quantization results by using (b) SAM, (c) SOM, and (d) BCL, respectively (e) MSE vs. neuron used plots

Cluster Number Estimation. In the second simulation, we use a three-dimensional data, as shown in Fig. 6(a). The data sets are difficult to cluster correct by the conventional competitive learning schemes. Fig. 6(b) shows the results after clustering via SAM. We can see that the network structure of SAM can fit in with input data distribution without any space projection. The link between neurons that belong to differrent class will be divided up without any post-process or clustering again.

| (a) Data Sample | (b) 27 neurons and 2 classes |
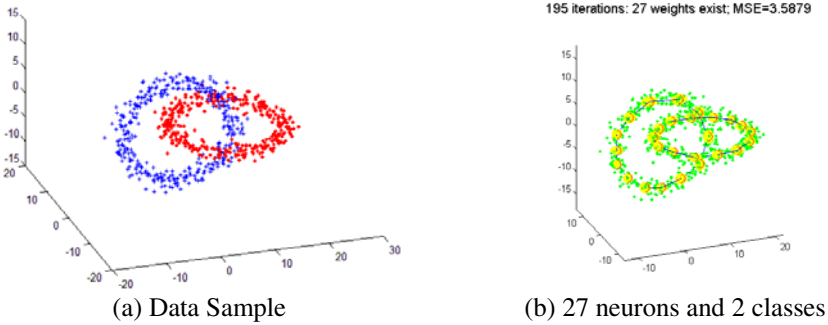
Fig. 6. Two cross rings data and clustering by SAM with parameters $(p, d, k_{merge}, k_{con})$ being set as (0.9, 1.5, 2, 2.5)

4 Conclusion

In this paper, we present a new competitive learning neural network with a special self-adjusting mechanism. It not only combines the advantages of self-creating, self-merging and self-deleting model but also uses local data statistics to dynamically adjust the neighborhood of representative neurons according to the spatial distribution of input data. The network size and topological structure of SAM can adaptively change according to input data distribution. From experimental results on simulations and real world applications, SAM presents better clustering or quantization results. Besides, SAM can appropriately estimate the cluster number in a data set and adaptively respond to various data input, even for some peculiar cases. To summarize, SAM is a novel adaptable neural network and it is recommended to be utilized as a prior process in various data involved.

References

1. Kohonen, T.: Self Organization and Associative Memory, 2nd edn. Springer Ser. Inform, vol. 8. Springer, New York (1984)
2. Kangas, J., Kohnen, T., Laaksonen, J.: Variants of self-organizing maps. IEEE Trans. Neural Networks 1, 93–99 (1990)

3. Rumelhart, D.E., McClelland, J.L.: The PDP research group: Parallel Distributed Processing, vol. 1, pp. 151–193. MIT Press, Cambridge (1986)
4. Mulier, F., Cherkassky, V.: Statistical analysis of self-organization. Neural Networks 8(5), 717–727 (1995)
5. Choi, D.-I., Park, S.-H.: Self-creating and Organizing Neural Networks. IEEE Trans. Neural Network 5(4), 561–575 (1994)
6. Fritzke, B.: Growing cell structures a self-organizing neural networks for unsupervised and supvised learning. Neural Networks 7(9), 1441–1460 (1994)
7. Fritzke, B.: A growing neural gas network learns topologies. In: Advances in Neural Information Processing Systems, vol. 7, pp. 625–632. MIT Press, Cambridge (1995)
8. Wang, J.-H., Sun, W.-D.: Online learning vector quantization: A harmonic competition approach based on conservation network. IEEE Trans. Syst. Man, Cybern.-Part B: Cybern. 29, 642–653 (1999)
9. Xiong, H., Ahmad, M.O., King, I.: Branching competitive learning Network: A novel self-creating model. IEEE Trans. Neural Networks 15, 417–429 (2004)
10. Martinetz, M., Schulten, K.J.: A "neural-gas" network learns topologies. In: Kohonen, K., Simula, O., Kangas, J. (eds.) Artif. Neural Netw., pp. 397–402. North-Holland, Amsterdam (1991)
11. Matsuyama, Y.: Harmonic competition: A self-organizing multiple criteria optimization. IEEE Trans. Neural Networks 7, 652–668 (1996)
12. Ritter, H., Martinetz, T., Schulten, K.: Neural Computation and Self-Organizing Map: An Introduction. Addison-Wesley, Reading (1992)
13. Brugger, D., Bogdan, M., Rosenstiel, W.: Automatic Cluster Detection in Kohonen's SOM. IEEE Trans. Neural Networks 19(3), 442–459 (2008)
14. Yin, H.: ViSOM: A novel method for multivariate data projection and structure visualization. IEEE Trans. Neural Networks 13(1), 237–243 (2002)
15. Wu, S., Chow, T.W.S.: PRSOM: a new visualization method by hybridizing multidimensional scaling and self-organizing map. IEEE Trans. Neural Networks 16(6), 1362–1380 (2005)
16. Yen, G.G., Wu, Z.: Ranked Centroid Projection: A Data Visualization Approach With Self-Organizing Maps. IEEE Trans. Neural Networks 19(2) (2008)
17. Tasdemir, K., Merenyi, E.: Exploiting Data Topology in Visualization and Clustering of Self-Organizing Maps. IEEE Trans. Neural Networks 20(4), 549–562 (2009)
18. Fritzke, B.: A self-organizing network that can follow nonstationary distributions. In: Gerstner, W., Hasler, M., Germond, A., Nicoud, J.-D. (eds.) ICANN 1997. LNCS, vol. 1327, pp. 613–618. Springer, Heidelberg (1997)
19. Datta, A., Pal, T., Parui, S.K.: A modified self-organizing neural net for shape extraction. Neurocomputing 14(1), 3–14 (1997)

Statistical Nonparametric Bivariate Isotonic Regression by Look-Up-Table-Based Neural Networks

Simone Fiori

Dipartimento di Ingegneria dell'Informazione
Facoltà di Ingegneria, Università Politecnica delle Marche
Via Brecce Bianche, Ancona I-60131, Italy
`s.fiori@univpm.it`
http://web.dibet.univpm.it/fiori

Abstract. Bivariate regression allows inferring a model underlying two data-sets. We consider the case of regression from possibly incomplete data sets, namely the case that data in the two sets do not necessarily correspond in size and might come unmatched/unpaired. The paper proposes to tackle the problem of bivariate regression through a nonparametric neural-learning method that is able to match the statistics of the available data sets. The devised neural algorithm is based on a look-up-table representation of the involved functions. A numerical experiment, performed on a real-world data set, serves to illustrate the features of the proposed statistical regression procedure.

Keywords: Statistical nonparametric regression; Bivariate isotonic regression; Look-up tables; Incomplete data sets; Nonparametric modeling.

1 Introduction

Statistical regression is advantageous in those applications where incomplete as well as complete data are available, which might also be unmatched. Data being incomplete means that specific observations are either lost or are not recorded exactly. Data happen to be unmatched/unpaired when they are recorded independently.

In some circumstances, the model underlying the data is known to be monotonic. An interesting example arises in forensics: A study on the modification of fingerprints due to ageing reveals that the mean distance between points-of-interests on fingerprints increases monotonically with age [7]. Any regression method devised under the hypothesis of monotonicity is referred to as 'isotonic regression' in statistics [12]. A widely invoked assumption on data is that missing data are missing at random [5,8]. The basic hypothesis underlying the missing-at-random scenario is that the pattern of incomplete data is traceable or predictable from other variables in the data-sets.

A natural application of regression is imputation of missing data, which consists in filling-in the 'holes' within the data. In missing-data handling approaches,

one of the desired outcomes is maintaining as closely as possible the shape of the original data distribution.

Missing-data handling techniques find widespread applications in medicine [3,15], social sciences [6], education [10], psychology [13] and meteorology [14]. More generally, applications in which uncertainty in the measurements make classical regression techniques appear unreliable, may take advantage of statistical regression technique. An example may be found in composite materials' mechanical properties characterization [2].

In the present paper, we consider tackling the problem of bivariate isotonic statistical regression via a neural look-up-table-based method that is able to match the statistics of the available data sets. A non-linear-system input/output statistical warping equation is interpreted as a valid regression equation for isotonic modelling. In effect, in classical input-output system analysis, the system transference is known and statistical methods are taken advantage of in order to describe the distribution of system output data on the basis of the distribution of the system input data. In the present context, the problem is reversed as the distribution of the input and output data are known, while a monotonic neural transference function that describes the warping of input data distribution into output data distribution needs to be identified.

Let us denote by $\mathcal{D}_x, \mathcal{D}_y$ the pair of data-sets a relationship among which is sought for. Regression is given here an interpretation as pooled-statistic matching problem for a learnable non-linear system. Namely, instead of considering variables $x \in \mathcal{D}_x$ and $y \in \mathcal{D}_y$ as paired and to look for a non-linear model that fits the variables values the best, we consider only cumulative information that arise by pooling the values within data-sets $\mathcal{D}_x$ and $\mathcal{D}_y$. As a consequence, the proposed regression technique allows to cope with the modelling problem when the size of the two data-sets do not match and/or when the pairing relationship of the values within $x \in \mathcal{D}_x$ and $y \in \mathcal{D}_y$ is unknown.

A key point of the method discussed here is that the quantities of interest as well as the designed model are represented in terms of paired lists of real numbers implemented as look-up-tables, which provide an efficient way of representing and handling the quantities of interest, as already shown in previous contributions published in the neural-network literature (see, e.g., paper [4] for an application to blind signal processing). An advantage offered by the devised procedure is the relieving from heavy computational requirement owing, e.g., to kernel methods (for a recent review, see, e.g., [11]).

ORGANIZATION OF THE PAPER: In Section 2, we discuss the isotonic regression problem in details and present the related analytic setting and its analytic solution. The analytic solution readily gives rise to an algorithmic formulation. We then describe the required operations with look-up tables and present the numerical procedure of statistical regression. In order to illustrate the behavior of the developed statistical regression technique, in Section 3 we consider a case of study. Section 4 concludes the paper.

2 Bivariate Isotonic Regression Method

We suppose that the neural regression model to learn has a non-linear structure described by relationship $y = f(x)$, where $x \in \mathcal{D}_x \subseteq \mathcal{R}$ denotes the input variable, having probability density function $p_x(\cdot)$, and $y \in \mathcal{D}_y \subseteq \mathcal{R}$ denotes the output variable, having probability density function $p_y(\cdot)$.

2.1 Properties of Statistical Bivariate Isotonic Regression

In the hypothesis that the neural regression model be *strictly monotonic*, namely $f'(x) > 0$ or $f'(x) < 0$, $\forall x \in \mathcal{D}_x$, the input-output distributions and the neural model transference may be shown to stay in the relationship:

- **Positive-slope neural model:** $f(x) = P_y^{-1}(P_x(x))$,
- **Negative-slope neural model:** $f(x) = P_y^{-1}(1 - P_x(x))$,

where symbol $P_y^{-1}(\cdot)$ denotes the inverse of the cumulative distribution function $P_y(\cdot)$ pertaining to the y-variable and symbol $P_x(\cdot)$ denotes the cumulative distribution function pertaining to the x-variable. In effect, the above transformations make sure that the distribution $p_x(\cdot)$ is transformed into the distribution $p_y(\cdot)$ according to the law of measure-invariance of probability density functions [9].

The problem of selecting a monotonically increasing or a monotonically decreasing type neural model is solvable by reasoning on the nature of the physical phenomena underlying the involved data-sets, regardless of the regression procedure.

The above regression method does not depend explicitly on the data, but on the probability density/cumulative functions obtained from each data set separately by pooling the data. Namely, data-sets may be incomplete (i.e., of different sizes) and the data may come unmatched.

2.2 Look-Up Tables and Required Elementary Operations

In order to develop a fully-numerical neural statistical regression method, the following ingredients are of use: A suitable numerical estimation method for cumulative density functions and a suitable format for function representation/handling (with particular emphasis on numerical function inversion). In order to put the above regression equations into effect, we chose to make use of a representation of the quantities of interest based on look-up-tables.

A real-valued look-up table with N entries is basically a pair LUT = $(\mathbf{x}, \mathbf{y})$, where $\mathbf{x} \in \mathcal{R}^N$ and $\mathbf{y} \in \mathcal{R}^N$. The entries x_k of vector $\mathbf{x}$ and the entries y_k of vector $\mathbf{y}$, with $k \in \{1, \ldots, N\}$, are paired and provide a point-wise description of an arbitrarily-shaped function. In order to handle the look-up tables for statistical modeling purpose, the following operations are of use:

- **Histogram Computation:** In order to numerically approximate the probability distribution of a data set $\mathcal{D}$, a histogram operator is of use. The

histogram-computation operation is denoted by $(\mathbf{x}, \mathbf{y}) = \mathrm{hist}(\mathcal{D})$. The constructed look-up table is built up as follows: x_k equals the value of the kth bin center, y_k equals the number of data-points falling in the kth bin.

– **Cumulative-Sum Computation:** On the basis of a look-up table $(\mathbf{x}, \mathbf{y})$, a new look-up table $(\mathbf{x}, \mathbf{v}) = \mathrm{csum}(\mathbf{x}, \mathbf{y})$ may be constructed, where array $\mathbf{v}$ contains the cumulative sum of the entries of array $\mathbf{y}$, possibly normalized in order to approximate the numerical integration of the function represented by the look-up table $(\mathbf{x}, \mathbf{y})$. In its un-normalized version, the cumulative sum is described by $v_1 = y_1$ and $v_k = v_{k-1} + y_k$ for $2 \leq k \leq N$.

– **Function/Table Interpolation:** Interpolation may be invoked to make computations with look-up tables on other points in the domain. In the present context, it is necessary to preserve the monotonicity of an approximated function, therefore we refer here to linear interpolation only. Let us denote by $\mathcal{D}$ the x-coordinate point-set, where the function represented by a look-up table $(\mathbf{x}, \mathbf{y})$ needs to be interpolated. The interpolation operation may be denoted by $\mathcal{I} = \mathrm{interp}(\mathbf{x}, \mathbf{y}, \mathcal{D})$, where the set $\mathcal{I}$ contains the interpolated y-values corresponding to the x-values in the set $\mathcal{D}$. Note that, because of the hypotheses of monotonicity of the model underlying the data, a set-type representation is equivalent to an ordered-list representation.

– **Function/Table Inversion:** If a function is given a point-wise representation by the help of a look-up table $(\mathbf{x}, \mathbf{y})$, then its inverse function may be easily given a point-wise representation by the look-up table $(\mathbf{y}, \mathbf{x})$, namely, function inversion is equivalent to swapping look-up table's arguments. *Apparently, therefore, function inversion in the context of look-up tables representation is a computationally-costless operation.*

2.3   Bivariate Isotonic Regression Algorithm

The neural bivariate isotonic regression method proposed here consists of the following steps.

First, it is necessary to estimate the probability density functions of data within $\mathcal{D}_x$ and $\mathcal{D}_y$ data-sets. They may be numerically estimated – up to scale factors – by:

$$(\mathbf{x}, \mathbf{p}_x) = \mathrm{hist}(\mathcal{D}_x), \ \ (\mathbf{y}, \mathbf{p}_y) = \mathrm{hist}(\mathcal{D}_y). \tag{1}$$

The numerical cumulative distribution functions of the $\mathcal{D}_x$ and $\mathcal{D}_y$ data-sets may now be estimated by numerical integration of the numerical probability density functions, which may be achieved by the help of the cumulative-sum operator applied to look-up tables $(\mathbf{x}, \mathbf{p}_x)$ and $(\mathbf{y}, \mathbf{p}_y)$. Namely, we construct look-up tables:

$$(\mathbf{x}, \mathbf{P}_x) = \mathrm{csum}(\mathbf{x}, \mathbf{p}_x), \ \ (\mathbf{y}, \mathbf{P}_y) = \mathrm{csum}(\mathbf{y}, \mathbf{p}_y). \tag{2}$$

If the statistical model has negative slope, the look-up table $(\mathbf{x}, \mathbf{P}_x)$ should be replaced by $(\mathbf{x}, 1 - \mathbf{P}_x)$ in what follows.

For regression purpose, the neural model is to be evaluated on a ordered set of x-points denoted here as an array $\hat{\mathbf{x}}$. Here, the array $\hat{\mathbf{x}}$ consists of R points

equally spaced over the interval of interest for the x-variable. The last step consists of numerically evaluating the quantity $P_x(\cdot)$ over the points in $\hat{\mathbf{x}}$ and then of evaluating the function $P_y^{-1}(\cdot)$ over the values of $P_x(\hat{\mathbf{x}})$, namely:

$$\mathcal{P}_x = \mathrm{interp}(\mathbf{x}, \mathbf{P}_x, \hat{\mathbf{x}}), \ \hat{\mathbf{y}} = \mathrm{interp}(\mathbf{P}_y, \mathbf{y}, \mathcal{P}_x). \tag{3}$$

Note that, in the second of equations (3), the inverse function $P_y^{-1}(\cdot)$ appears through the swapped look-up table of function $P_y(\cdot)$.

The pair $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ provides a look-up table representation for the non-linear model $f(\cdot)$.

3   Numerical Experiment with a Real-World Data-Set

In order to illustrate the features of the proposed statistical regression procedure, in the present section we show and comment on the results of a numerical experiment performed on a real-world data set.

Data used here were collected by D. Donoho and E. Ramos in 1982 and were used during the 1983 Data Exposition of the American Statistical Association
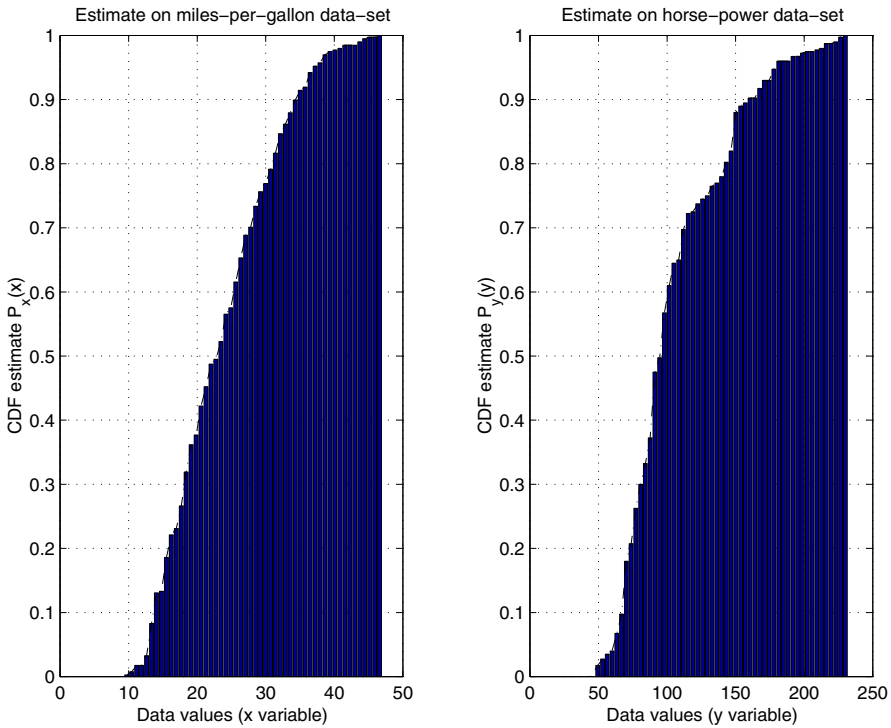
Fig. 1. Cumulative distribution functions of the data-sets $\mathcal{D}_x$ and $\mathcal{D}_y$ required by the regression procedure
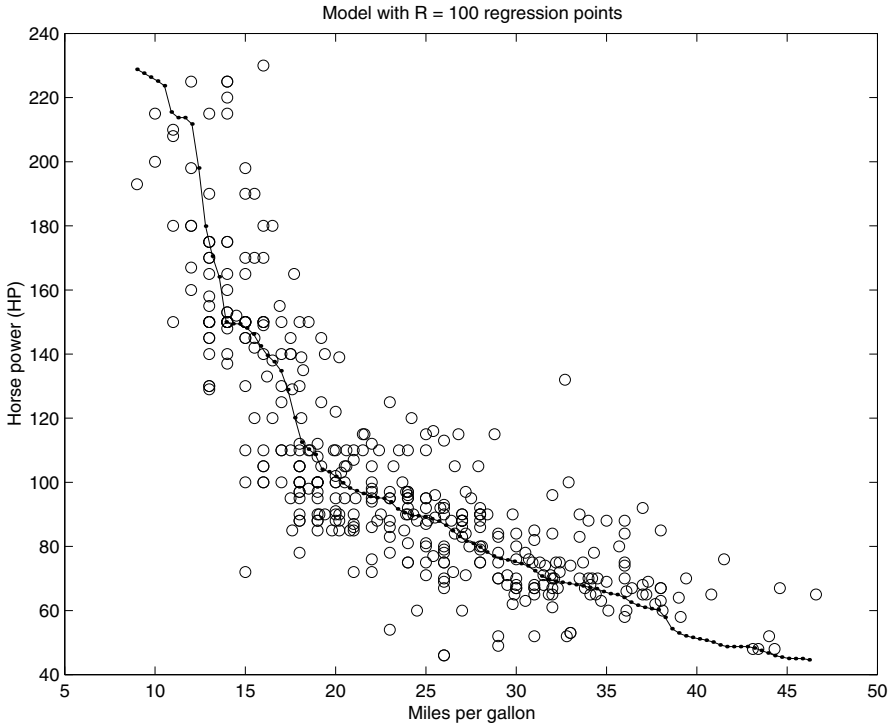
Fig. 2. Computed regression model superimposed on data-cloud. (Solid-dotted line: Regression curve. Open circles: Data-points obtained by leaving out incomplete records from original data.)

(ASA). Data deal with automobiles. Measures on automobile features, such as fuel consumption (miles per gallon), number of cylinders and engine displacement were provided. The data-set is available, e.g., from the ASA web-page [1].

In the experiment, we used fuel efficiency (miles-per-gallon, MPG) as well as car's power (horse-power, HPW) as variables a relationship among which is sought for. Both data sets are incomplete and 'holes' do not match across data-sets, so we regarded such data-sets as unpaired.

The cumulative distribution functions of the MPG-variable and HPW-variable, required by the regression procedure, are depicted in the Figure 1.

Intuitively, the greater the power of an automobile (HPW), the more its fuel consumption, hence the less its fuel efficiency (MPG). Such observation leads to the selection of a monotonically decreasing regression model.

The regression model obtained by launching the neural bivariate isotonic regression procedure devised in the present paper is depicted in the Figure 2. The shown regression model was obtained by setting $R = 100$ points as regression precision. In order to obtain a fair visualization of the modeling capability of

the proposed bivariate isotonic regression method, the Figure 2 also shows the original data points obtained by leaving out incomplete records.

As it is readily seen, even if the data-cloud is rather spread, the obtained regression curve appears to fit the data in a quite satisfactory way (the computed regression model looks like a 'skeletonization' of the data-cloud). No prior information on the model was made use of, except for the assumption on its (decreasing) monotonicity.

On data sets of about 400 data-points each, the regression procedure takes about 16 milliseconds to run on a 1.86GHz, 256MB platform under MATLAB$^\copyright$ environment.

4   Conclusion

It is known that real-world systems and phenomena cannot be accurately described by a mathematical model to be evaluated analytically. In this case, statistical regression provides a useful tool to build up a model of the phenomenon under observation. In the present paper, we discussed the problem of isotonic bivariate regression via a learnable neural system that is able to match its input-output statistic to the statistic of available data-sets, a relationship among which is sought for.

In the present paper, we proceeded under the missing-at-random assumption for the incomplete data case. Regression was given here an interpretation as pooled-statistic matching problem for a non-linear system. Also, the assumption that the regression model should be monotonically increasing/decreasing was made, which led to the development of a bivariate isotonic regression method.

In order to get a better insight into the neural statistical regression method proposed in the present paper, a numerical experiment performed on a real-world data-set was conducted. The result of such numerical experiment showed that the computed model fits the data in a satisfactory way.

We are currently seeking for an extension of the discussed isotonic bivariate regression approach to the many-to-one-variable case. The main challenge here is how to formulate the multivariate regression approach by statistics matching. From the implementation side, the extension of traditional (two-variable) look-up tables to multi-variable look-up tables appears straightforward.

References

1. American Statistical Association, http://www.amstat.org
2. Biagiotti, J., Fiori, S., Torre, L., López-Manchado, M.A., Kenny, J.M.: Mechanical Properties of Polypropylene Matrix Composites Reinforced with Natural Fibers: A Statistical Approach. Polymer Composites 25(1), 26–36 (2004)
3. Cook, N.R.: Imputation Strategies for Blood Pressure Data Nonignorably Missing Due to Medication Use. Clinical Trials 3(5), 411–420 (2006)
4. Fiori, S.: Hybrid Independent Component Analysis by Adaptive LUT Activation Function Neurons. Neural Networks 15(1), 85–94 (2002)

5. Ghahramani Z., Jordan M.I.: Learning from Incomplete Data. A.I. Memo No. 1509 of the Artificial Intelligence Laboratory, and C.B.C.L. Paper No. 108 of the Center for Biological and Computational Learning Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology (1994)
6. Greve, H.R., Brandon Tuma, N., Strang, D.: Estimation of Diffusion Processes from Incomplete Data (A Simulation Study). Sociological Methods & Research 29(4), 435–467 (2001)
7. Gottschlich, C., Hotz, T., Lorenz, R., Bernhardt, S., Hantschel, M., Munk, A.: Modeling the Growth of Fingerprints Improves Matching for Adolescents. IEEE Transactions on Information Forensics and Security (August 9, 2010), http://arxiv.org/abs/1008.1150, Preprint of the aXiv archive (to appear)
8. Little, R.J.A., Rubin, D.A.: Statistical Analysis with Missing Data. John Wiley and Sons, New York (1987)
9. Papoulis, A.: Probability and Statistics. Prentice-Hall (1996)
10. Peugh, J.L., Enders, C.K.: Missing Data in Educational Research: A Review of Reporting Practices and Suggestions for Improvement. Review of Educational Research 74(4), 525–556 (2004)
11. Pozdnoukhov, A., Bengio, S.: Semi-Supervised Kernel Methods for Regression Estimation. In: Proceedings of the International Conference on Acoustics, Speech and Signal Processing (IEEE-ICASSP 2006), Toulouse, France, vol. 5 (2006)
12. Ramsay, J.O.: Estimating Smooth Monotone Functions. Journal of the Royal Statistical Society B 60, 365–375 (1998)
13. Roth, P.: Missing Data: A Conceptual Review for Applied Psychologists. Personnel Psychology 47, 537–560 (1994)
14. Schneider, T.: Analysis of Incomplete Climate Data: Estimation of Mean Values and Covariance Matrices and Imputation of Missing Values. Journal of Climate 14, 853–871 (2001)
15. Wood, A.M., White, I.R., Thompson, S.G.: Are Missing Outcome Data Adequately Handled? A Review of Published Randomized Controlled Trials in Major Medical Journals. Clinical Trials 1(4), 368–376 (2004)

Recovery of Sparse Signal
from an Analog Network Model

Chi-Sing Leung[1], John Pui-Fai Sum[2], Ping-Man Lam[1], and A.G. Constantinides[3]

[1] Dept. of Electronic Engineering, City University of Hong Kong, Hong Kong
[2] Institute of Electronic Commerce, National Chung Hsing University
[3] Imperial College, UK
`eeleungc@cityu.edu.hk, pfsum@nchu.edu.tw`

Abstract. This paper presents an analog neural network model to recover sparse signals. In the original constrained optimization task for recovering sparse signals, the objective function is not differentiable. Hence, we recast the original nonlinear programming problem as a linear programming problem with linear inequality constraints and equality constraints. However, the second order gradient of the objective function is not convex at an equilibrium point. To solve this problem, we further modify the objective function such that the second order gradient is convex at the equilibrium point. This paper presents two sets of network dynamics. One is for the standard recovery of sparse signals. Another one is for the noisy situation.

Keywords: Optimization, Sparse signal.

1 Introduction

Sparse sampling is a sampling technique in data acquisition for sparse signals [1, 2]. The theory in sparse sampling tells us that a sparse signal can be represented by a few measured values. To obtain those measured values, the sparse signal is measured by a set of pseudo random functions. Traditional approaches to recover sparse signals from sparse sampling are based on Newton's method.

The analog computational circuit approach is one of important methods for solving optimization problems in the neural networks community. In the past three decades, a number of analog neural circuits for optimization have been proposed [3, 4]. The advantage of the analog neural approach is that the computation is completed in massively parallel architecture. In [5], a canonical circuit was proposed to solve nonlinear programming problems with inequality constraints. Wang [6, 7] used the concept of dual neural networks to solve a number of problems, including constrained quadratic programming problems. Apart from optimization, neural circuits can also be used for searching the maximum of a set of numbers [7, 8].

In [9, 10], the Lagrange programming neural network (LPNN) model was proposed to solve general constrained optimization problems. The model is based on the Lagrange multiplier theory in optimization. The LPNN model aims at finding a solution satisfying the necessary conditions of optimality. The network consists of two types of neurons: variable and Lagrange neurons. The variable neurons store the variables being optimized. The Lagrange neurons store the Lagrange multipliers. The role of variable

neurons is to decrease the Lagrange function while the role of Lagrange neurons is to guide the state of the network into the feasible region.

This paper adopts the LPNN model to recover sparse signal. Since the objective function for recovering sparse signal is not differentiable, we recast the original nonlinear programming problem as a linear programming problem with linear inequality constraints and equality constraints. However, the second order gradient of the Lagrange function at an equilibrium point is not convex. To solve this problem, we modify the objective function by introducing an augmented approach such that the second order gradient is convex. We propose the LPNN dynamics to handle two situations, including the standard recovery of sparse signals and the noisy measurement values.

2   Background

In sparse sampling, a sparse signal $x \in \Re^n$ is measured by a set of m random vectors, $\{\phi_1, \cdots, \phi_m : \phi_j \in \Re^n\}$, where $m < n$. The m measured values are given by

$$y_1 = \langle x, \phi_1 \rangle, \ \cdots, y_m = \langle x, \phi_m \rangle, \tag{1}$$

where $\langle \cdot, \cdot \rangle$ is the inner product operator. In the matrix-vector form, the measured signal is given by $y = \Phi x$. Theoretically, ϕ_i's are global support random vectors. Practically, we can use pseudo random valued vectors, such as noiselets [2]. The advantage of using noiselets is that there is an efficient algorithm to computation the measurement values. The sparse signal can be recovered from the measured signal y by solving the following constrained optimization problem, given by

$$\min \ \|x\|_{l_1} \ \text{subject to} \ \ y = \Phi x. \tag{2}$$

Theoretically, the signal can be recovered when the number of non-zero elements is less than $C_o \log n$, where C_o is a constant.

Let $x \in \Re^n$ be the variables being optimized. A LPNN aims at minimizing the following constrained optimization problem:

$$\text{Minimize} \ f(x) \ \text{subject to} \ h(x) = 0, \tag{3}$$

where $f(x)$ is the objective function, $h(x) : \Re^n \to R^m$ describes the m equality constraints, and $m < n$. The LPNN model uses the Lagrange multiplier approach to obtain the optimized solution of (3). The Lagrange function is given by

$$\mathcal{L}(x, \lambda) = f(x) + \lambda^T h(x), \tag{4}$$

where $\lambda = [\lambda_1, \cdots, \lambda_m]^T$ is the Lagrange multiplier vector. In the LPNN approach, we use kinds of neurons, variable and Lagrange neurons, to realize the optimization. The variable neurons are seeking for a minimum point of (3) and the Lagrange neurons are trying to constrain the state of the system. The neural dynamics is given by

$$\frac{dx}{dt} = -\nabla_x \mathcal{L}(x, \lambda) \ \text{and} \ \frac{d\lambda}{dt} = \nabla_\lambda L(x, \lambda). \tag{5}$$
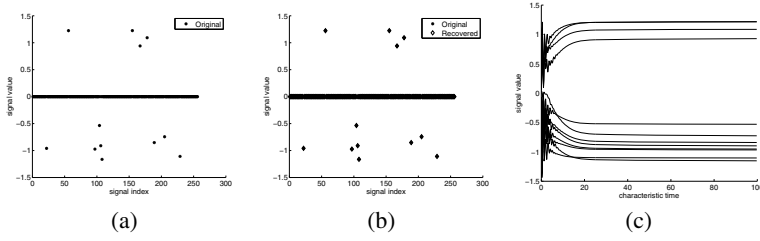
Fig. 1. Signal Recovery of 1D signal. (a) The original signal. (b) Recovery signal. (c) The dynamic of the non-zero signal values.

In (5), the equilibrium point $(\boldsymbol{x}^*, \boldsymbol{\lambda}^*)$, given by $\frac{d\boldsymbol{x}}{dt}|_{(\boldsymbol{x}^*, \boldsymbol{\lambda}^*)} = 0$ and $\frac{d\boldsymbol{\lambda}}{dt}|_{(\boldsymbol{x}^*, \boldsymbol{\lambda}^*)} = 0$, fulfills the following conditions:

$$\nabla_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}^*) = \nabla f(\boldsymbol{x}^*) + \nabla h(\boldsymbol{x}^*) \boldsymbol{\lambda}^{*T} = \boldsymbol{0} \tag{6}$$

$$\nabla_{\boldsymbol{\lambda}} \mathcal{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}^*) = \nabla h(\boldsymbol{x}^*) = \boldsymbol{0}. \tag{7}$$

The dynamics (5) seeks to provide solutions satisfying the necessary conditions of optimality. The equilibrium point of the network satisfies the Kuhn-Tucker condition for the problem. The stability of the network are guaranteed under some general convexity conditions [9, 10]. Let $(\boldsymbol{x}^*, \boldsymbol{\lambda}^*)$ be an equilibrium point of $L(\boldsymbol{x}, \boldsymbol{\lambda})$ of the network. If $\nabla_{\boldsymbol{x}}^2 \mathcal{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}^*) > 0$ and $\nabla_{\hat{\boldsymbol{x}}} h(\hat{\boldsymbol{x}}) = \{\nabla_{\hat{\boldsymbol{x}}} h_1(\hat{\boldsymbol{x}}), \cdots, \nabla_{\hat{\boldsymbol{x}}} h_m(\hat{\boldsymbol{x}})\}$ of $h(\hat{\boldsymbol{x}})$ are linear independent, $(\boldsymbol{x}^*, \boldsymbol{\lambda}^*)$ is a stable point.

3 Recovery for Sparse Signal

To recover a sparse signal from the measured vector $\boldsymbol{y}$ by LPNNs, one may suggest that we can define the objective function of the LPNN model as

$$\mathcal{L}(\hat{\boldsymbol{x}}, \boldsymbol{\lambda}) = \|\boldsymbol{x}\|_{l_1} + \boldsymbol{\lambda}^T (\boldsymbol{y} - \boldsymbol{\Phi} \boldsymbol{x}). \tag{8}$$

However, the norm–1 measure $\| \cdot \|_{l_1}$, involving the absolute operator $|x|$, is not differentiable at $x = 0$. Instead of giving approximation for the absolute operator, we recast the nonlinear programming problem (2) as the following programming problem:

$$\min \sum_i^n w_i, \text{ subject to } 0 \geq x_i - w_i, \ 0 \geq -x_i - w_i \ \forall i, \ \boldsymbol{y} = \boldsymbol{\Phi} \boldsymbol{x}. \tag{9}$$

Define $\boldsymbol{z} = (\boldsymbol{w}^T, \boldsymbol{x}^T)^T$. (9) can be rewritten as

$$\min \boldsymbol{q}^T \boldsymbol{z}, \text{ subject to } \boldsymbol{0} \geq \boldsymbol{A} \boldsymbol{z}, \ \boldsymbol{y} = \boldsymbol{A}_{eq} \boldsymbol{z}, \text{ where} \boldsymbol{A} = \begin{pmatrix} -\boldsymbol{I}_n & \boldsymbol{I}_n \\ -\boldsymbol{I}_n & -\boldsymbol{I}_n \end{pmatrix}, \tag{10}$$

$$\boldsymbol{A}_e q = [\varnothing | \boldsymbol{\Phi}], \ \boldsymbol{q}^T = (\underbrace{1, \cdots, 1}_{n}, \underbrace{0, \cdots, 0}_{n}), \tag{11}$$

and $\boldsymbol{I}_n$ is an identity matrix. We can change the inequality constraints to equality constraints by introducing stack variables $\{s_1, \cdots, s_{2n}\}$. The problem becomes

$$\min \boldsymbol{q}^T \boldsymbol{z}, \ \text{subject to } \boldsymbol{A}\boldsymbol{z} + \left(s_1^2, \cdots, s_{2n}^2\right)^T = \boldsymbol{0}, \ \boldsymbol{y} = \boldsymbol{A}_{eq}\boldsymbol{z}. \tag{12}$$

To simplify the notation, we introduce the pointwise multiplication operator $\otimes$ of two vectors, where $\boldsymbol{a} \otimes \boldsymbol{b} = (a_1 b_1, a_2 b_2, \cdots)^T$. Now, (12) becomes

$$\min \boldsymbol{q}^T \boldsymbol{z}, \ \text{subject to } \boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s} = \boldsymbol{0}, \ \boldsymbol{y} = \boldsymbol{A}_{eq}\boldsymbol{z}. \tag{13}$$

One may think that we can use the LPNN approach to solve (13). However, the second order gradient of the corresponding Lagrange function is equal to zero and thus the convexity condition does not hold. So, an equilibrium point $(\boldsymbol{x}^*, \boldsymbol{\lambda}^*)$ is not stable.

To ensure the local convexity [10], we introduce the an augmented term $\frac{c_o}{2}(\boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s})^T(\boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s})$ in (13), where c_o is a constant. Thus, (13) becomes

$$\min \boldsymbol{q}^T \boldsymbol{z} + \frac{c_o}{2}\|\boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s})\|^2, \ \text{subject to } \boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s} = \boldsymbol{0}, \ \boldsymbol{y} = \boldsymbol{A}_{eq}\boldsymbol{z}. \tag{14}$$

At an equilibrium point of the neural dynamics, "$\boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s}$" is equal to zero (please see (18)). Hence, there is no change on the objective function at the equilibrium point.

Now, the Lagrange function of (14) is given by

$$\mathcal{L}(\boldsymbol{z}, \boldsymbol{s}, \boldsymbol{\lambda}, \boldsymbol{\beta}) = \boldsymbol{q}^T \boldsymbol{z} + \frac{c_o}{2}\|\boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s}\|^2 + \boldsymbol{\lambda}^T(\boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s}) + \boldsymbol{\beta}^T(\boldsymbol{y} - \boldsymbol{A}_{eq}\boldsymbol{z}). \tag{15}$$

In (15), $\boldsymbol{z}$, $\boldsymbol{s}$, $\boldsymbol{\xi}$ and p are state variables, while $\boldsymbol{\lambda}$ and $\boldsymbol{\beta}$ are Lagrange variables. With (15), the dynamics of the network is given by

$$\frac{d\boldsymbol{z}}{dt} = -\boldsymbol{q} - c_o(\boldsymbol{A}^T \boldsymbol{A}\boldsymbol{z} + \boldsymbol{A}^T(\boldsymbol{s} \otimes \boldsymbol{s})) - \boldsymbol{A}^T \boldsymbol{\lambda} + \boldsymbol{A}_{eq}^T \boldsymbol{\beta} \tag{16}$$

$$\frac{d\boldsymbol{s}}{dt} = -2\boldsymbol{s} \otimes (\boldsymbol{\lambda} + c_o(\boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s})) \tag{17}$$

$$\frac{d\boldsymbol{\lambda}}{dt} = \boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s}, \ \frac{d\boldsymbol{\beta}}{dt} = \boldsymbol{y} - \boldsymbol{A}_{eq}\boldsymbol{z}. \tag{18}$$

From (11), matrix $\boldsymbol{A}$ only contains identify matrices and matrix $\boldsymbol{A}_{eq}$ contains a zero matrix. So, the dynamics (16)-(18) do not involve $2n \times 2n$ matrix computation. It can be rewritten as

$$\frac{d\boldsymbol{w}}{dt} = -\begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} - c_o \left(2\boldsymbol{w} + \begin{pmatrix} -s_1^2 - s_{n+1}^2 \\ \vdots \\ -s_n^2 - s_{2n}^2 \end{pmatrix} \right) - \begin{pmatrix} -\lambda_1 - \lambda_{n+1} \\ \vdots \\ -\lambda_n - \lambda_{2n} \end{pmatrix} \tag{19}$$

$$\frac{d\boldsymbol{x}}{dt} = -c_o \left(2\boldsymbol{x} + \begin{pmatrix} s_1^2 - s_{n+1}^2 \\ \vdots \\ s_n^2 - s_{2n}^2 \end{pmatrix} \right) - \begin{pmatrix} \lambda_1 - \lambda_{n+1} \\ \vdots \\ \lambda_n - \lambda_{2n} \end{pmatrix} + \boldsymbol{\Phi}^T \boldsymbol{\beta} \tag{20}$$

$$\frac{d\boldsymbol{s}}{dt} = -2\boldsymbol{s} \otimes \left(\boldsymbol{\lambda} + c_o \left(\begin{pmatrix} -\boldsymbol{w} + \boldsymbol{x} \\ -\boldsymbol{w} - \boldsymbol{x} \end{pmatrix} + \boldsymbol{s} \otimes \boldsymbol{s} \right) \right), \ \frac{d\boldsymbol{\lambda}}{dt} = \begin{pmatrix} -\boldsymbol{w} + \boldsymbol{x} \\ -\boldsymbol{w} - \boldsymbol{x} \end{pmatrix} + \boldsymbol{s} \otimes \boldsymbol{s} \tag{21}$$

$$\frac{d\boldsymbol{\beta}}{dt} = \boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}. \tag{22}$$

In the above, only (20) and (22) involve matrix-vector multiplication. In (20), we need to compute the product of a $n \times m$ matrix and a m dimension vector. In (22), we need to compute the product of a $m \times n$ matrix and a n dimension vector. That means the number of weights is $O(m \times n)$. In this sense, we can consider that the dynamics is a special form of bi-directional associative memories (BAMs) [11]. There are two major layers of neurons. One layer stores the state variables $\boldsymbol{x}$. Another layer stores the Lagrange variables $\boldsymbol{\beta}$. From (20) and (22), the connection matrix provides the information flow between the state variables and the Lagrange variables. Practically, the weight matrix $\boldsymbol{\Phi}$ can be a random $\{+1, -1\}$ matrix. This suggests that the implementation of the analog neural circuit could be very simple because we have no need to precisely construct the weight matrix.
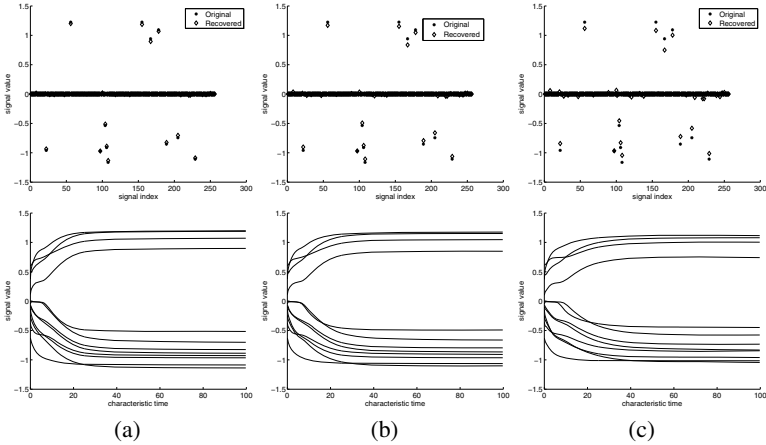
Fig. 2. Signal Recovery of 1D signal from noisy measurement. The first row is the recovery signal. The second row is the dynamic of the non-zero signal values. (a) measurement noise variance $\sigma^2 = 0.0025$. (b) measurement noise variance $\sigma^2 = 0.01$. (c) measurement noise variance $\sigma^2 = 0.04$.

4   Recovery from Noisy Measured Signal

This section considers the situation that the measured signal $\boldsymbol{y}$ contains noise, given by

$$\boldsymbol{y} = \boldsymbol{\Phi x} + \boldsymbol{\xi} \,, \tag{23}$$

where $\boldsymbol{\xi}$ are independently identical random noise with zero mean and variance σ^2. So, the residual $\|\boldsymbol{y} - \boldsymbol{\Phi x}\|^2$ should be less than the noise power. That means, $\boldsymbol{y} - \boldsymbol{\Phi x} = \boldsymbol{\xi}$ and $\boldsymbol{\xi}^T \boldsymbol{\xi} \leq m\sigma^2$. Introducing a stack variable p, we can have $\boldsymbol{y} - \boldsymbol{\Phi x} = \boldsymbol{\xi}$ and $\boldsymbol{\xi}^T \boldsymbol{\xi} - m\sigma^2 + p^2 = 0$. So, our aim is to solve the following problem:

$$\text{Minimize } \|\boldsymbol{x}\|_{l_1} \text{ subject to } \boldsymbol{y} - \boldsymbol{\Phi x} = \boldsymbol{\xi}, \boldsymbol{\xi}^T \boldsymbol{\xi} - m\sigma^2 + p^2 = 0. \tag{24}$$

Again, the gradient of the objective function is not differentiable. Hence, we recast the nonlinear programming problem (24) as the following programming problem:

$$\min \boldsymbol{q}^T \boldsymbol{z}, \text{ subject to } \boldsymbol{0} \geq \boldsymbol{A}\boldsymbol{z}, \ \boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x} = \boldsymbol{\xi}, \boldsymbol{\xi}^T\boldsymbol{\xi} - m\sigma^2 + p^2 = 0. \quad (25)$$

Converting the inequality constraints to equality constraints, we have

$$\min \boldsymbol{q}^T \boldsymbol{z}, \text{ subject to } \boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s} = \boldsymbol{0}, \ \boldsymbol{y} - \boldsymbol{A}_{eq}\boldsymbol{z} - \boldsymbol{\xi} = 0, \boldsymbol{\xi}^T\boldsymbol{\xi} - m\sigma^2 + p^2 = 0. \quad (26)$$

To maintain the convexity, the augmented term is given by $\frac{c_o}{2}\|\boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s}\|^2$. So the augmented Lagrange function is given by

$$\mathcal{L}(\boldsymbol{z}, \boldsymbol{s}, \boldsymbol{\xi}, p, \boldsymbol{\lambda}, \boldsymbol{\beta}, \mu) = \boldsymbol{q}^T\boldsymbol{z} + \frac{c_o}{2}\|\boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s}\|^2 + \boldsymbol{\lambda}^T(\boldsymbol{A}\boldsymbol{z} + \boldsymbol{s} \otimes \boldsymbol{s})$$
$$+ \boldsymbol{\beta}^T(\boldsymbol{y} - \boldsymbol{A}_{eq}\boldsymbol{z} - \boldsymbol{\xi}) + \mu(\boldsymbol{\xi}^T\boldsymbol{\xi} - m\sigma^2 + p^2). \quad (27)$$

In (27), $\boldsymbol{z} = (\boldsymbol{w}^T, \boldsymbol{x}^T)^T$, $\boldsymbol{s}$, $\boldsymbol{\xi}$ and p are state variables, while $\boldsymbol{\lambda}$, $\boldsymbol{\beta}$, and μ are Lagrange variables. With the Lagrange function, the dynamics is given by

$$\frac{d\boldsymbol{w}}{dt} = -\begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} - c_o \left(2\boldsymbol{w} + \begin{pmatrix} -s_1^2 - s_{n+1}^2 \\ \vdots \\ -s_n^2 - s_{2n}^2 \end{pmatrix} \right) - \begin{pmatrix} -\lambda_1 - \lambda_{n+1} \\ \vdots \\ -\lambda_n - \lambda_{2n} \end{pmatrix} \quad (28)$$

$$\frac{d\boldsymbol{x}}{dt} = -c_o \left(2\boldsymbol{x} + \begin{pmatrix} s_1^2 - s_{n+1}^2 \\ \vdots \\ s_n^2 - s_{2n}^2 \end{pmatrix} \right) - \begin{pmatrix} \lambda_1 - \lambda_{n+1} \\ \vdots \\ \lambda_n - \lambda_{2n} \end{pmatrix} + \boldsymbol{\Phi}^T\boldsymbol{\beta} \quad (29)$$

$$\frac{d\boldsymbol{s}}{dt} = -2\boldsymbol{s} \otimes \left(\boldsymbol{\lambda} + c_o \left(\begin{pmatrix} -\boldsymbol{w} + \boldsymbol{x} \\ -\boldsymbol{w} - \boldsymbol{x} \end{pmatrix} + \boldsymbol{s} \otimes \boldsymbol{s} \right) \right), \ \frac{d\boldsymbol{\xi}}{dt} = \boldsymbol{\beta} - 2\mu\boldsymbol{\xi}, \quad (30)$$

$$\frac{d\boldsymbol{\lambda}}{dt} = \begin{pmatrix} -\boldsymbol{w} + \boldsymbol{x} \\ -\boldsymbol{w} - \boldsymbol{x} \end{pmatrix} + \boldsymbol{s} \otimes \boldsymbol{s}, \ \frac{dp}{dt} = -2\mu p, \ \frac{d\mu}{dt} = \boldsymbol{\xi}^T\boldsymbol{\xi} - m\sigma^2 + p \quad (31)$$

$$\frac{d\boldsymbol{\beta}}{dt} = \boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}. \quad (32)$$

Again, only (29) and (32) involve matrix-vector multiplication. That means the number of weights is $O(m \times n)$. The dynamics can be considered as a special form of bi-directional associative memories (BAMs) [11]. One layer stores the state variables $\boldsymbol{x}$. Another layer stores the Lagrange variables $\boldsymbol{\beta}$. From (29) and (32), the connection matrix provides the information flow between the state variables and the Lagrange variables.

5   Simulation Results

The first example is an artificial 1D signal, shown in Figure 1(a). The signal contains 256 samples, of which 244 samples are of zero value while 12 samples are of non-zero values. These non-zero values are randomly distributed. Figure 1(b) shows the recovery of the 1D sparse artificial signal from 60 measured values with no measurement noise. From the figure, the recovered signal has the nearly same to the original signal. Moreover, the recovered signal has converged after 40 characteristic times, as shown in Figure 1(c).
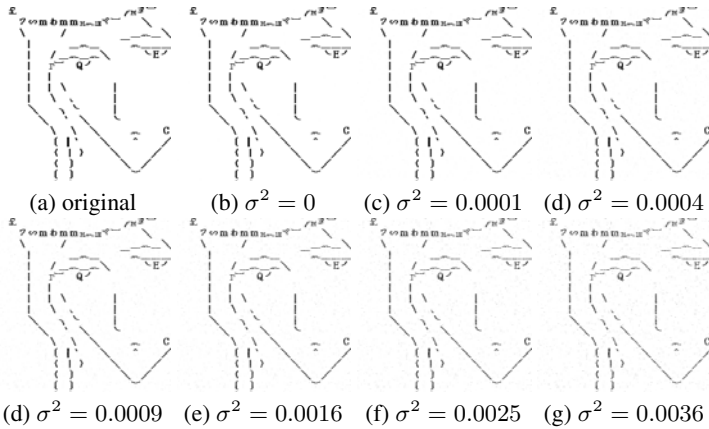
Fig. 3. Recovered Images for "Queen" for various measurement noise levels
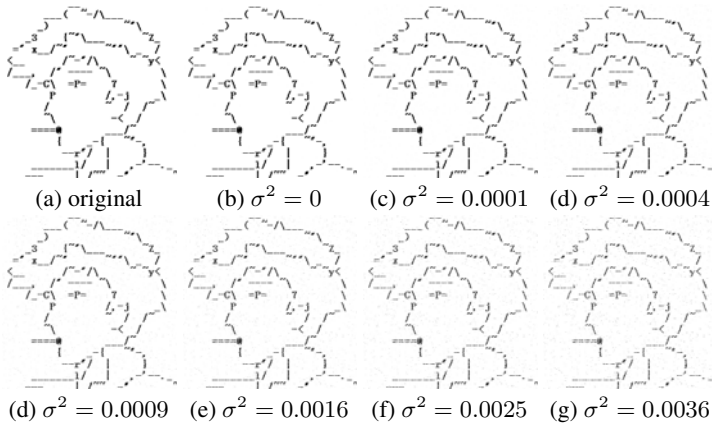

Fig. 4. Recovered Images for "Boy" for various measurement noise levels

Figures 1 shows the recovery of the 1D sparse artificial signal from 60 noisy measured values. The noise added to the measured values is independently identical random noise with zero mean. We consider three values of noise variance $\sigma^2 = \{0.0025, 0.01, 0.04\}$. In this experiment, the values of recovered signal are also close to the original signal as shown in Figure 2(a)-(c). The recovered signal is also converged after 40 characteristic times, as shown in Figure 2(d)-(f).

We further use two binary images with resolution of $256 \times 256 = 65536$ pixels to verify our LPNN model. The images, shown in Figure 3(a) and 4(a), are sparse. $15,000$ measured values are used for recovering. For image "Queen", the average power of the measured values is equal to 0.0326. For image "Boy", the average power of the measured values is equal to 0.0386. We add Gaussian measurement noise to the measured values. The noise variances are given by

$$\sigma^2 = \{0.0, 0.0001, 0.0004, 0.0009, 0.0016, 0025, 0.0036\}.$$

Figure 3 and Figure 4 show the recovered images. From the figure, the LPNN method can recover the image when there are noise in the measured values. As expected, the recovered images are degraded when the variance of the noise increases.

6   Conclusion

In this paper, we formulate the LPNN model to handle the signal recovery in sparse sampling. Hence, the objective function of sparse sampling contains a non- differentiable term, we recast the problem as a linear programming problem with equality constraints. However, the linear programming approach cannot lead the local convexity condition in the LPNN. Hence, we further modify the objective function based on the augmented approach. We propose the LPNN dynamics to handle two situations, including the recovery of sparse signal and the noisy measurement values. Simulation results verify that our approach can be applied for recovering 1D and 2D signals in sparse sampling for both noise free and noisy environments.

References

1. Candès, E.J., Wakin, M.B.: An introduction to compressive sampling. IEEE Signal Processing 25(1), 21–30 (2008)
2. Coifman, R., Geshwind, F., Meyer, Y.: Noiselets. Applied and Computational Harmonic Analysis 10, 27–44 (2001)
3. Tank, D.W., Hopfield, J.J.: Simple neural optimization networks: An AD converter, signal decision circuit, and a linear programming circuit. IEEE Trans. Circuits and Systems 33, 533–541 (1986)
4. Culioli, J.C., Protopopescu, V., Britton, C.L., Ericson, M.N.: Neural networks models for linear programming. In: Proc. IJCNN, vol. 1, pp. 293–296 (1990)
5. Chua, L., Lin, G.: Nonlinear programming without computation. IEEE Trans. on Circuits Syst. 31, 182–188 (1984)
6. Liu, S., Wang, J.: A Simplified Dual Neural Network for Quadratic Programming With Its KWTA Application. IEEE Transactions on Neural Networks 17(6), 1500–1510 (2006)
7. Hu, X., Wang, J.: An Improved Dual Neural Network for Solving a Class of Quadratic Programming Problems and Its k-Winners-Take-All Application. IEEE Transactions on Neural Networks 19(12), 2022–2031 (2008)
8. Sum, J., Leung, C.S., Tam, P., Young, G., Kan, W., Chan, L.W.: Analysis for a class of winner-take-all model. IEEE Trans. Neural Networks 10(1), 64–71 (1999)
9. Zhang, S., Constantinidies, A.G.: Lagrange programming neural networks. IEEE Trans. on Circuits and Systems II 39, 441–452 (1992)
10. Zhu, X., Constantinidies, A.G.: Lagrange programming neural networks for Linear Programming. Journal of Parallel and Distributed Computing 14, 354–360 (1993)
11. Kosko, B.: Bidirectional associative memories. IEEE Trans. Syst. Man Cybern. 18(1), 49–60 (1988)

A VLSI Spiking Neural Network with Symmetric STDP and Associative Memory Operation

Frank L. Maldonado Huayaney[1], Hideki Tanaka[1], Takayuki Matsuo[1], Takashi Morie[1], and Kazuyuki Aihara[2]

[1] Graduate School of Life Science and Systems Engineering, Kyushu Institute of Technology, Kitakyushu 808-0196, Japan
[2] Institute of Industrial Science, University of Tokyo 4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan

Abstract. This paper proposes an analog CMOS VLSI circuit which implements integrate-and-fire spiking neural networks with spike-timing dependent synaptic plasticity (STDP). The designed VLSI chip includes 25 neurons and 600 synapse circuits with symmetric all-to-all connection STDP. Using the fabricated VLSI chip, we implement a Hopfield-type feedback network, and demonstrate its associative memory operation. In our chip, analog information is represented by the relative timing of spike firing events. Symmetric STDP provides an auto-correlation learning function depending on relative timing between spikes consisting of a learning pattern. Each learning and test pattern consists of 20 spike pulses each of which has a relative delay corresponding to a gray-scale pixel intensity. The chip has successfully associated from an input pattern the most similar learning pattern.

Keywords: VLSI, spiking neural network, STDP, associative memory.

1 Introduction

Spiking neuron models, which express analog information by the timing of neuronal spike firing, attract a lot of attention with expectation of their higher information processing ability [2,3]. From the viewpoint of VLSI implementation, spiking neurons output binary values in the voltage or current domain and represent analog values in the time domain. Therefore, spiking neural network systems can easily be connected to the existing digital systems than the conventional analog neural network systems that output analog values in the voltage or current domain.

Some silicon circuits and VLSI chip for spiking neurons with asymmetric spike-timing dependent synaptic plasticity (STDP) were reported [5,1]. We have demonstrated that the integrate-and-fire (IF) neuron model can be applied to the Hopfield network, which is a typical feedback network model, and that it has a retrieval property as associative memory [6]. Furthermore, we proposed an analog CMOS circuit for IF neurons with STDP, and demonstrated memorization by patterns using symmetric STDP and retrieval of the pattern in a Hopfield-type feedback spiking neural network [7].

In this paper, we describe our designed VLSI chip of spiking neural networks with STDP based on the circuit developed in our previous work, and show measurement results about associative memory operation using the fabricated VLSI chips. The measurement results demonstrate that the VLSI chip realizes the symmetric STDP functions and spiking feedback network operation with learning by STDP.

2   VLSI Spiking Neural Network

2.1   Spiking Feedback Network Model with GEU

In the IF neuron model, if the internal state value does not exceed the threshold, the neuron does not generate spikes. However, in order to apply the spiking neuron model to feed-back networks with continuous states, spikes have to be generated pseudo-periodically to express an analog value by spike timing. To achieve this operation, we have proposed a global excitatory unit (GEU) [6].

GEU receives spikes from all neurons via excitatory synapses, and it is activated by the earliest input. The activated GEU gives a continuous level stimulus to all neurons, therefore even a neuron at the resting state can fire, and a neuron with stronger inhibition generates a spike with later timing.

2.2   CMOS Neuron and Synapses Circuits

Our CMOS spiking neuron circuit and its timing diagram are shown in Fig. 1. The CMOS spiking neuron circuit consists of a synapse part and a neuron part. If spike pulse i_i is fed into the synapse part from other neurons, a PSP control signal (*psp_cont*) is generated by a delay-and-inversion circuit (*D&I*) and a NOR gate. While *psp_cont* is "High", transconductance amplifier (gm-amp) A_1 turns on and charges or discharges capacitor C_{int} in the neuron part. Thus, the capacitor terminal voltage V_n, which is the internal state of the neuron, is changed, and a PSP is generated. The spatiotemporal summation of PSPs by input spikes is performed at this capacitor. The current from the synapse is determined by $V_{w_{ij}}$. The gm-amp A_1 generates a current in proportion to $V_{w_{ij}} - V_{ref}$. When $V_{w_{ij}} - V_{ref} > 0$, the circuit operates as excitatory synapse, and vice versa.

In the neuron part, the internal state potential represented by V_n returns to resting potential V_{ini} by leak resistance R_L connected in parallel with capacitor C_{int}, after charged or discharged by the current sources of the synapse parts. A comparator (*CMP*) compares V_n with threshold voltage V_{th}. If V_n exceeds threshold voltage V_{th}, a spike is generated. At the same time, the threshold voltage increases to generate a refractory period. In this circuits, the transmission delay is equal to the refractory period.

2.3   Symmetric STDP Circuit

Figure 2(a) and (b) show our CMOS symmetric STDP circuit and its timing diagram, respectively. The circuit consists of a spike detection part (*SD*) and a
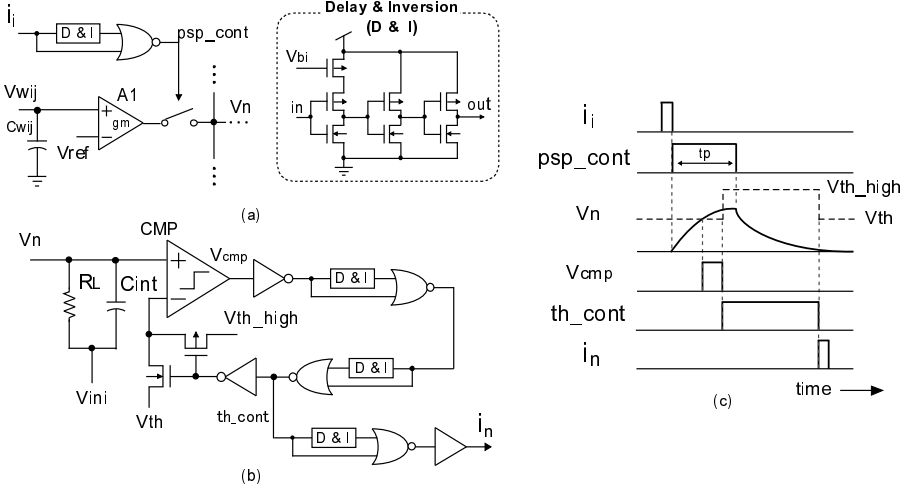
Fig. 1. Spiking neuron circuit: (a) synapse part (PSP generation), (b) neuron part and (c) timing diagram

weight update part (*WU*), as shown in Fig. 2(a). Since the circuit operation is symmetric with respect to the order of input timing of spikes *pre* and *post*, the circuit works only with the absolute time difference of *pre* and *post*. To achieve this operation, *SD* inverts the order of input spikes or forwards them depending on which of them arrives first.

In *SD*, the state value of the toggle flip-flop (T-FF) is changed twice by input spikes *pre* and *post*. Changes in the state value are detected by a *D&I* and a NOR gate. As a result, an earlier spike pulse is fed into *in1* of *WU*, and the other is fed into *in2*. If *pre* and *post* are given at the same time, the state value of the T-FF changes once. In that case, the T-FF is reset by an AND gate, a *D&I* and a NOR gate.

Weight update part *WU* updates the synaptic weight represented by $V_{w_{ij}}$ based on the time difference between input spikes *in1* and *in2*. Generation of an STDP function is achieved using a current sampling scheme [4]; the first spike *in1* triggers generation of a nonlinear waveform $V_{NW}(t)$ that has the same shape as the STDP function, and the second spike *in2* samples the current generated using $V_{NW}(t)$. The detail of the operation is as follows. When the input spike *in1* is fed into *WU* from *SD*, ramp signal $V_A(t)$ is generated by charging capacitance C_A with a constant current by MOSFET M_A. At the same time, control signal V_{SW}, which is generated by the *D&I*, turns to "High". Ramp signal $V_A(t)$ is transformed to a nonlinear waveform by MOSFET M_B and capacitor C_B. After V_{SW} turns to "Low", the capacitor terminal voltage V_{NW} returns to reference voltage V_{ref} by resistor R. Thus, the shape of $V_{NW}(t)$ is the same as the half part of the symmetric STDP function, as shown in Fig. 2(b). The gm-amp $A2$ updates synaptic weight capacitor voltage $V_{w_{ij}}$ with current $i(t)$ generated by $G_m(V_{NW}(t) - V_{ref})$, where G_m is the transconductance; $\Delta V_{w_{ij}}$ is given by the
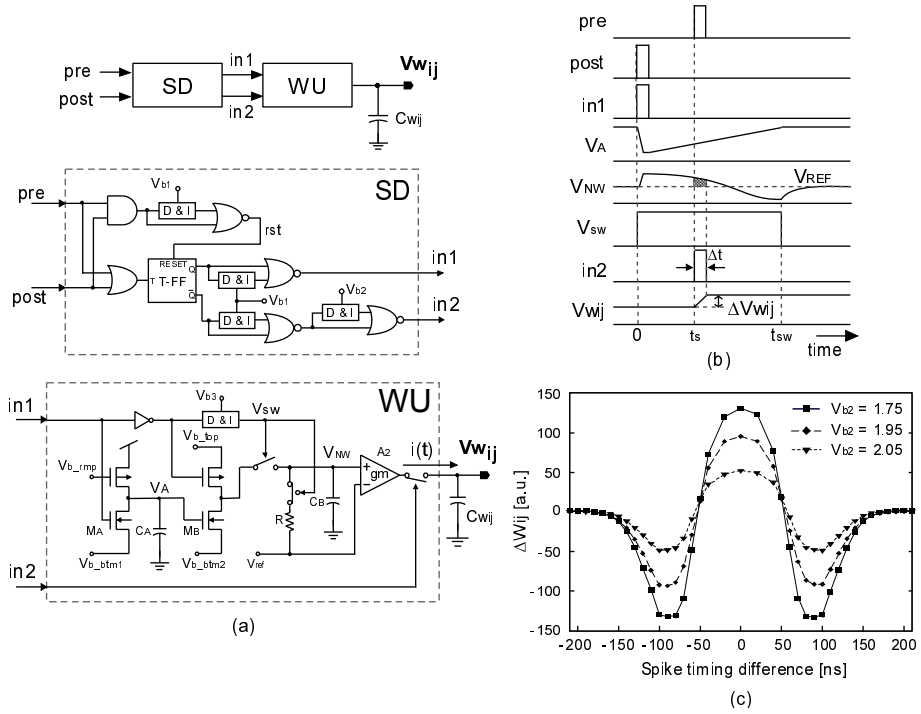
Fig. 2. Symmetric STDP circuit: (a) circuit, and (b) timing diagram, and (c) measurement results of symmetric STDP function with a test chip

integral of $i(t)$ during the time span $[t_s, t_s + \Delta t]$. Symmetric STDP characteristics measured using a test chip are shown in Fig. 2(c) [7].

3   VLSI Chip Design and Measurement Results

3.1   VLSI Chip Architecture and Design

We designed a CMOS VLSI chip by using TSMC 0.25 μm (1-Poly, 5-Metal) CMOS technology. The chip includes 25 neuron circuits and 600 synapse circuits with symmetric STDP function and those with asymmetric STDP, which are not described in this paper. Synapse circuits with symmetric STDP comprises symmetric all-to-all connections. The chip architecture is shown in Fig. 3, and the chip photograph and specification are shown in Fig. 4.

3.2   Measurement Condition

We measured the output response of the Hopfield-type feedback network consisting of 20 neurons trained with two learning patterns. Data is expressed by the relative timing of asynchronous spike firing events. we define a grey scale
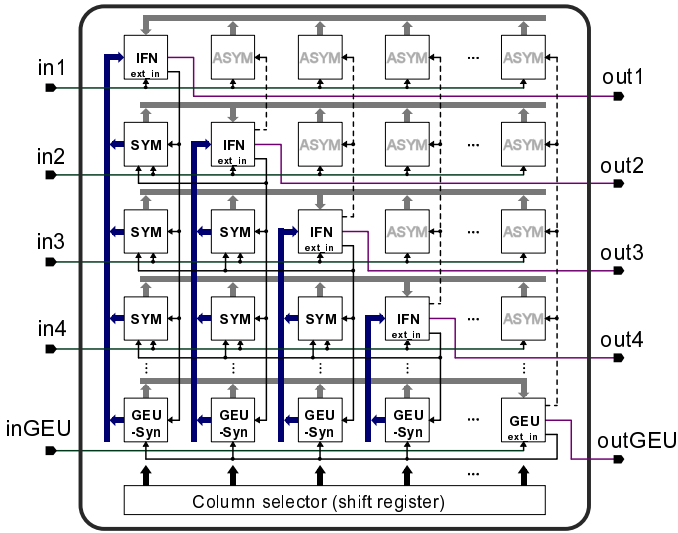
Fig. 3. Chip architecture: neuron (IFN), synapse with symmetric STDP (SYM), synapse with asymmetric STDP (ASYM), and GEU

Specification		
Technology		TSMC 0.25 μm
Number of neurons		25
Number of synapses with sym. STDP		600
Number of synapses with asym. STDP		300
Operation period		200 ns
Power consumption*	Neuron part	280 μW/unit
	Synapse (sym) part	250 μW/unit
	Synpase (asym) part	100 μW/unit
		*HSPICE simulation
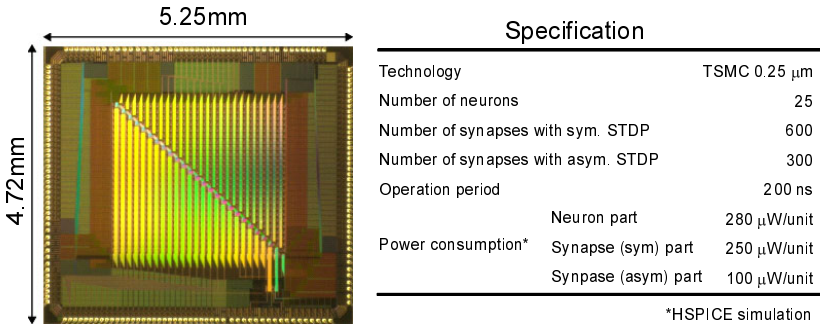
Fig. 4. Chip photograph and specification

timing from 0 to 100 ns, where 0 ns represent black and 100 ns white color, as shown in Fig. 5.

Our network was trained with the two learning (memorized) patterns shown in Fig. 5. As a similarity measure between patterns, we used the Manhattan distance D_M, which is defined by:

$$D_M = \sum_{i=1}^{N} |A_i - B_i| \qquad (1)$$

where A_i and B_i are elements of the respective patterns, and N is the number of elements. A_i (or B_i) is 0 for a black pixel and 1 for white. A value of $D_M = 0$
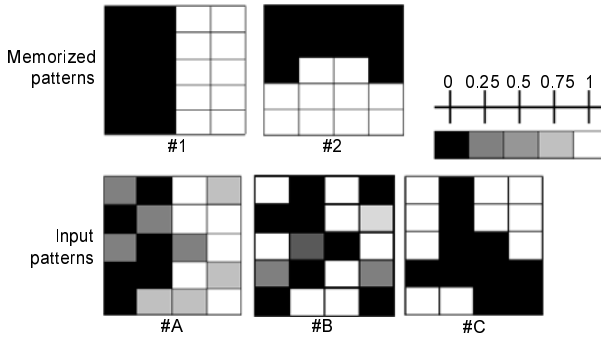
Fig. 5. Patterns for associative memory experiment: two learning (memorized) patterns and three input patterns

Table 1. Manhattan distances between memorized and input patterns used in the experiment

Manhattan distance	input #A	input #B	input #C
memorized #1 (original)	3 (min.)	7.5 (min.)	10
memorized #2 (original)	10.8	11.6	16
memorized #1 (equivalent)	14.2	12.6	10
memorized #2 (equivalent)	9.4	10	4 (min.)

means that the two patterns are identical, while a large value means that both patterns are very different. Note that the pattern reversed with black and white is considered as the equivalent with the original pattern in our associative memory.

The input patterns used in recall experiments are also shown in Fig. 5, and the Manhattan distances from the memorized patterns are shown in Table 1. The system tries to converge to the image with the minimum D_M.

3.3 Experiment Results

The association results are shown in Fig. 6. In all three cases, the results are the same as the theoretical results predicted. For input pattern #A ($D_M = 3$ to memorized pattern #1), as shown in Fig. 6(a), the circuit successfully converges to the memorized pattern #1. The elapsed time to find the first pattern was around 400 ns.

For input pattern #B ($D_M = 7.5$ to memorized pattern #1), as shown in Fig. 6(b), the circuit converges to the memorized pattern #2. In this case, since the input pattern is more different from that in the first case, the convergence time is longer, 1.8 μs.

For input pattern #C ($D_M = 4$ to memorized pattern #2), as shown in Fig. 6(c), the circuit converges to the memorized pattern #2. Similar to the first case, since D_M is small, the convergence time is also short.
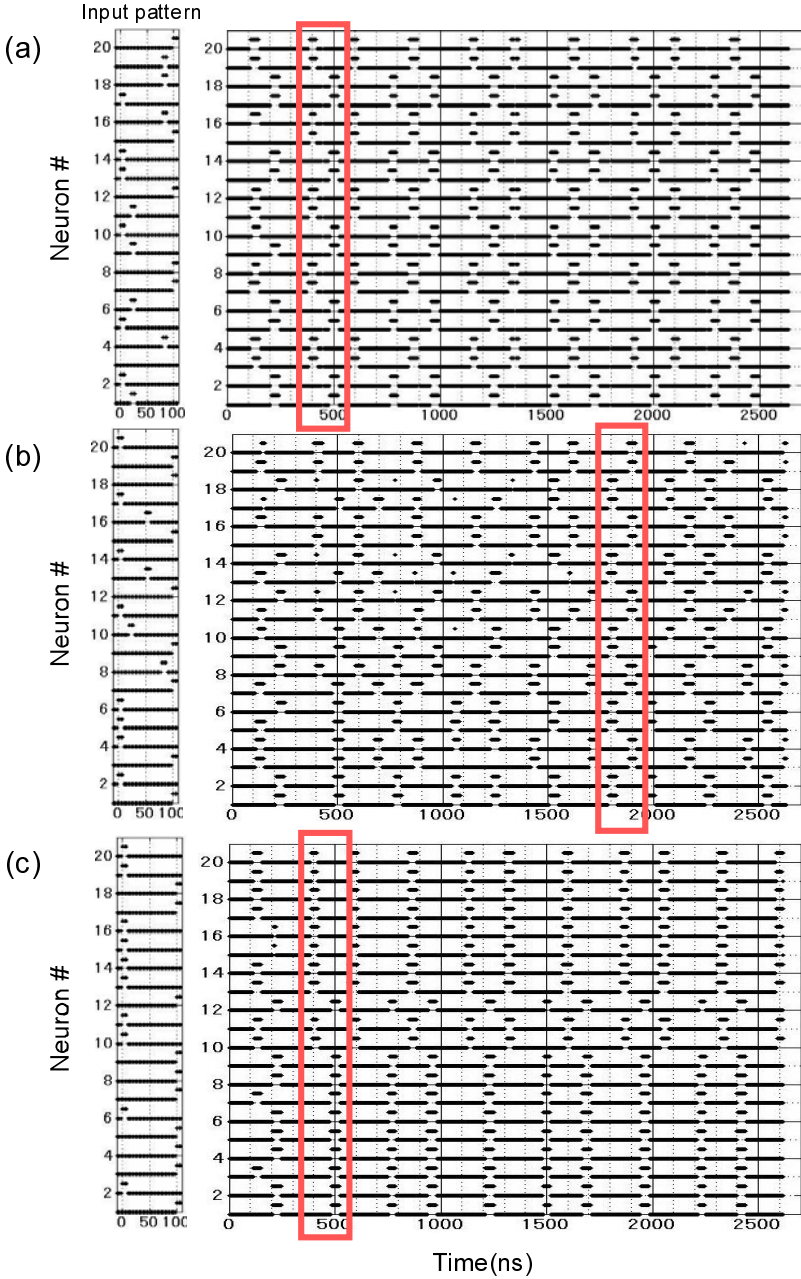
Fig. 6. Experimental results of recall operation in associative memory using fabricated chip: input pattern is #A (a), #B (b), and #C (c). Rectangles indicate convergence timing).

4   Conclusion

We designed, fabricated and tested a VLSI spiking neural network with symmetric STDP function. The measurement results for associative memory operation in a Hopfield-type feedback network demonstrated that our VLSI chip works successfully. Our chip learned two patterns by symmetric STDP, and recalled the closest memorized pattern for some input patterns, as the theory predicts using Manhattan distance. Our spiking associative memory system operates in high speed; it requires only two or three iterations for input patterns near a memorized pattern.

References

1. Indiveri, G., Chicca, E., Douglas, R.: A VLSI array of low-power spiking neurons and bistable synapses with spike-timing dependent plasticity. IEEE Trans. Neural Networks 17(1), 211–221 (2006)
2. Maass, W.: Networks of spiking neurons: The third generation of neural network models. Neural Networks 10(9), 1659–1671 (1997)
3. Maass, W., Bishop, C.M. (eds.): Pulsed Neural Networks. MIT Press, Cambridge (1999)
4. Morie, T., Murakoshi, K., Nagata, M., Iwata, A.: Pulse modulation techniques for nonlinear dynamical systems and a CMOS chaos circuit with arbitrary 1-D maps. IEICE Trans. Electron. E87-C(11), 1856–1862 (2004)
5. Bofill-i-Petit, A., Murray, A.F.: Synchrony detection and amplification by silicon neurons with STDP synapses. IEEE Trans. Neural Networks 15(5), 1296–1304 (2004)
6. Sasaki, K., Morie, T., Iwata, A.: A VLSI spiking feedback neural network with negative thresholding and its application to associative memory. IEICE Trans. Electronics 89-C(11), 1637–1644 (2006)
7. Tanaka, H., Morie, T., Aihara, K.: A CMOS spiking neural network circuit with symmetric/asymmetric STDP function. IEICE Trans. Fundamentals E92-A(7), 1690–1698 (2009)

Method of Solving Combinatorial Optimization Problems with Stochastic Effects

Takahiro Sota[1], Yoshihiro Hayakawa[2], Shigeo Sato[1], and Koji Nakajima[1]

[1] The Laboratory for Brainware/Laboratory for Nanoelectronics and Spintronics, Research Institute of Electrical Communication, Tohoku University 2-1-1 Katahira, Aoba-Ku, Sendai 980-8577, Japan sota@nakajima.riec.tohoku.ac.jp, {hello,shigeo}@riec.tohoku.ac.jp

[2] Department of Information Systems, Sendai National College of Technology 4-16-1 Ayashi-Chuoh, Aoba-Ku, Sendai 989-3128, Japan hayakawa@sendai-nct.ac.jp

Abstract. The higher order connections network is useful to solve the combinatorial optimization problems, however, the network topology is complicated so that implementation on hardware is not easy. To implement the higher order connections more simply, we introduce the stochastic logic architecture to the discrete hysteresis network with the higher order connections. The proposed network can solve a Traveling Salesman Problems as the conventional network.

Keywords: Higher order connections, Hysteresis neural network, Quartic form, Energy function, Traveling Salesman Problems.

1 Introduction

The combinatorial optimization problems such as the Traveling Salesman Problems (TSPs) or Quartic Assignment Problems (QAPs) include a lot of applications, for example, the packet routing or delivery planning. However, it is not easy to obtain useful solutions of these problems in real time by using conventional computers. As one of the solver of these problems, we have proposed the method to use the Inverse function Delayed network with the Higher order synaptic Connections (HC-ID network). The ID network is a neural network that has a negative resistance effect in its dynamics, and the output space of the network has a region of the negative resistance effect. The network state is updated along the gradient of the quadratic form energy function if the state is outside the region; otherwise the state is not updated along the gradient of the energy function. Hence we can destabilize the undesirable states through appropriate setting of the negative resistance [1].

In the HC-ID network, we expand the energy function to the quartic form by introducing the higher order connections. This network present a solution of the TSPs or QAPs, in which cost at a stationary stable stateis smaller than a prearranged cost [2]. Hence this method is a powerful solver of the combinatorial optimization problems, and we aim to implement the HC-ID network on chip to

obtain the solution in real time. However, it is not easy to implement the HC-ID network directly because the higher order connections are more complicated compare to the conventional synaptic connections.

In this paper, we introduce the stochastic logic to the higher order connections network. The stochastic logic is common technique to code analog values in the pulse stream. It performs the product-sum operation by using basic logic gates, then the higher order connections may be implemented more simply by introducing the logic. The stochastic logic has been introduced to the neural network with linear connections[3], however, it has not yet to the network with the higher order connections. We propose a network with the higher order connections to be operated by using the stochastic logic and investigate the performance of the proposed network to solve the TSPs through numerical experiments.

2   Solving Combinatorial Optimization Problems by the Higher Order Connections Network

The discrete hysteresis network with the higher order connection is derived from the HC-ID network by introducing the binary outputs and discrete time update[4]. This network can also solve the combinatorial optimization problems. In this section, we describe the discrete hysteresis network with the quartic form energy function applying it to solve the combinatorial optimization problems.

2.1   Discrete Hysteresis Network with Higher Order Connections

The discrete hysteresis network with the 3rd order connections is

$$
\delta_i = \frac{1}{\tau_u} \left(\sum_{j=1}^{N} \sum_{k=1}^{N} \sum_{l=1}^{N} w_{ijkl} X_j(t) X_k(t) X_l(t) \right.
$$
$$
\left. + \sum_{j=1}^{N} \sum_{k=1}^{N} w_{ijk} X_j(t) X_k(t) + \sum_{j=1}^{N} w_{ij} X_j(t) + h_i \right), \tag{1}
$$

$$
U_i(t+1) = \begin{cases} U_i(t) & \text{if } (X_i(t) = 0 \text{ and } U_i(t) < 0 \text{ and } \delta_i < 0) \\ & \quad \text{or } (X_i(t) = 1 \text{ and } U_i(t) > 0 \text{ and } \delta_i > 0), \\ U_i(t) + \delta_i & \text{otherwise} \end{cases} \tag{2}
$$

$$
X_i(t+1) = \begin{cases} 0 & \text{if } U_i(t+1) < -\alpha \\ 1 & \text{if } U_i(t+1) > \alpha \\ X_i(t) & \text{otherwise} \end{cases}, \tag{3}
$$

where N is the number of neurons, $U_i(t)$ and $X_i(t)$ are the internal state and output of a neuron i at time t, respectively, and h_i is bias of a neuron i. $w_{ijk\cdots}$ is the synaptic weight from neurons $j, k, \cdots$ to a neuron i. τ_u is the time constant. The parameter α is positive constraint which controls the depth of hysteresis, and if $\alpha = 0$ this network behaves as well as the discrete Hopfield network.

Because the updating of u is restricted as shown in (2), the internal states do not diverge.

The energy function of this network is

$$
\begin{aligned}
E_{\mathtt{HC-Hys}}\big(\boldsymbol{X}(t)\big) = & -\frac{1}{4\tau_u}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}\sum_{l=1}^{N} w_{ijkl} X_i(t) X_j(t) X_k(t) X_l(t) \\
& -\frac{1}{3\tau_u}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N} w_{ijk} X_i(t) X_j(t) X_k(t) \\
& -\frac{1}{2\tau_u}\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij} X_i(t) X_j(t) - \frac{1}{\tau_u}\sum_{i=1}^{N} h_i X_i(t),
\end{aligned}
\tag{4}
$$

where $\boldsymbol{X}(t)$ is the network state at time t. This energy function is 4th order (quartic) form function for the 3rd order connections.

2.2 Quartic Form Energy Function for Solving Combinatorial Optimization Problems

Let us consider an $n \times n$ neuron unit matrix. The quartic energy functions for various combinatorial optimization problems are designed as

$$
\begin{aligned}
E_{\mathtt{CH}}\big(\boldsymbol{X}\big) = & \frac{A}{2}\sum_{i=1}^{n}\left(\sum_{x=1}^{n} X_{xi} - 1\right)^2 + \frac{A}{2}\sum_{x=1}^{n}\left(\sum_{i=1}^{n} X_{xi} - 1\right)^2 \\
& + \frac{B}{2}\sum_{x=1}^{n}\sum_{i=1}^{n}\sum_{y=1}^{n}\sum_{j=1}^{n} b_{xi,yj} X_{xi} X_{yj}(1 - X_{xi} X_{yj}) \\
& + \frac{C}{2}\left(\sum_{x=1}^{n}\sum_{i=1}^{n}\sum_{y=1}^{n}\sum_{j=1}^{n} b_{xi,yj} X_{xi} X_{yj}\right)^2,
\end{aligned}
\tag{5}
$$

where A, B and C are positive coefficients and X_{xi} means the output of neuron unit (x,i). $b_{xi,yj} \in \boldsymbol{b}^{[n^2 \times n^2]}$ is a cost value when the neurons (x,i) and (y,j) fire $(X_{xi} \sim 1,\ X_{yj} \sim 1)$. The first, second and third terms of (5) denote constraint conditions. The first and second terms are constraints that take the minimum value 0 when only one neuron fires on each row and column. The third term will be zero when all outputs of the network are zero or one. If these constraints are satisfied, the forth term becomes the square value of the solution cost indicated by the network state.

When this energy function is applied to discrete hysteresis network, the network stability is denoted as

$$
\frac{B}{4C}
\begin{cases}
> c_{\mathtt{sol}}(\boldsymbol{x}) & \text{then the state } \boldsymbol{x} \text{ is stable} \\
< c_{\mathtt{sol}}(\boldsymbol{x}) & \text{then the state } \boldsymbol{x} \text{ is unstable}
\end{cases},
\tag{6}
$$

where $c_{\mathtt{sol}}(\boldsymbol{x})$ is the cost value of the network state $\boldsymbol{x}$. (6) indicates that we can obtain only the solutions in which costs are less than $B/4C$ at the stationary stable state.
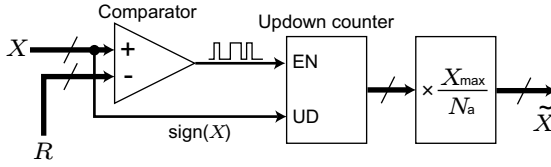
Fig. 1. Stochastic coding and decoding circuit

3   Introducing Stochastic Logic

The stochastic logic is a digital technique that realize pseudo-analog operations using stochastically coded pulse sequences[3]. The neuron hardware using the stochastic logic has high reliability thanks to digital circuits, and decreases the area consumption. In this section we introduce the stochastic logic to the discrete hysteresis network with the higher order connections.

3.1   Stochastic Logic

Figure 1 shows the stochastic coding and decoding circuits, where X is an analog value to be coded, $X_{\max}$ is the maximum value of X, and R is a random number which ranges $[0 : X_{\max}]$ uniformly. X and R are the inputs for the comparator, which outputs are 1 if $X > R$, or 0 else. The expectation and the variance of the accumulation of these pulse sequences are

$$E[\tilde{X}] = \frac{X_{\max}}{N_a} \cdot N_a P_f = X, \tag{7}$$

$$V[\tilde{X}] = \left(\frac{X_{\max}}{N_a}\right)^2 N_a P_f (1 - P_f) = \frac{1}{N_a} X (X_{\max} - X), \tag{8}$$

where N_a is the number of accumulation, P_f is the firing probability, and $\tilde{X}$ is an accumulated value normalized by $X_{\max}/N_a$. (8) implies that N_a controls the quantity of the coding noise.

3.2   Modification of Update Rules

The stochastic logic generates a stochastic noise in the product-sum operations, so that the update restriction of the internal state also has to take into account the effect of the noise. (2) is modified for the network with the stochastic logic as

$$U_i(t+1) = \begin{cases} U_i(t) & \text{if } (X_i(t) = 0 \text{ and } U_i(t) < 0 \text{ and } \delta_i < -f) \\ & \quad \text{or } (X_i(t) = 1 \text{ and } U_i(t) > 0 \text{ and } \delta_i > f), \\ U_i(t) + \delta_i & \text{otherwise} \end{cases} \tag{9}$$

where f means the permissible range. Because the distribution of the pulses follows the Gaussian distribution, we set $f = \max\left(\sqrt{V[\tilde{X}]}\right) = W_{\max}/(2\sqrt{N_a})$.
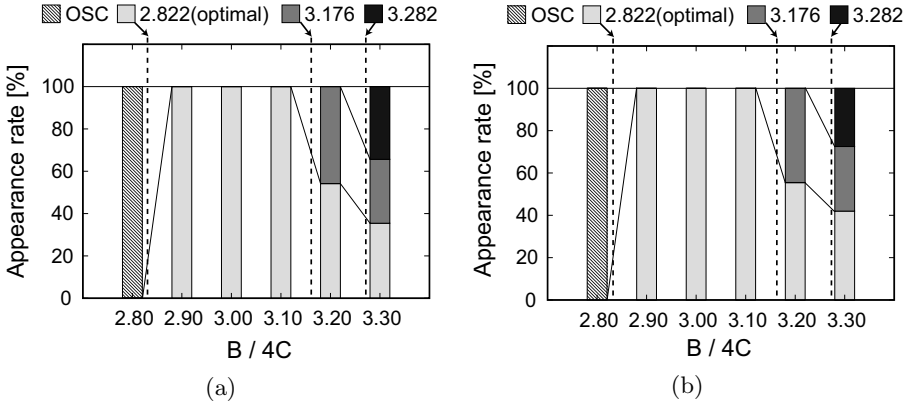
Fig. 2. Network state dependency on the parameter $B/4C$ of the conventional network (a) and the proposed network (b). The parameter sets are $A = 3.04$ and $C = 0.10$, and B is adjusted along the value of $B/4C$. The accumulate time N_a is 2.7×10^5 in (b). 'OSC' denotes a oscillation state.

4   Numerical Experiments Results

In this section we show the performance of the proposed network to solve the TSPs through numerical experiments. In all experiments, $\alpha = 0$, $\tau_u = 1$, and the parameter A of the energy function is set to satisfy the constraint conditions[2]. The solved problems are a 4-city TSP and 6-city TSP.

4.1   Comparing with Conventional Network

First, we compare the proposed network with the conventional network by solving the 4-TSP. Figure 2 shows the appearance rate of the network state depending on the parameter $B/4C$. The colors of bars in Fig. 2 denote the network states. The oscillation state means that the network does not become stable state without showing any solution, and the others are the stationary stable states. The proposed network can search the solutions similar to the conventional network if the number of accumulation is enough large.

4.2   Effect of the Number of Accumulation

Second, we investigate the relation between the number of accumulation N_a and the appearance states by solving the 6-TSP. The result is shown in Fig. 3. The noise caused by the stochastic pulse coding increases with decreasing N_a, and this noise destabilize the solution states in order of the cost from largest. When N_a is set appropriately, the network presents only the optimal solutions at stationary stable state even though the parameter $B/4C$ is not the optimal value. Hence for appropriate N_a the proposed network indicates better solutions compared to the conventional network. However, the stochastic noise destabilizes even the optimal solutions if N_a is too small.
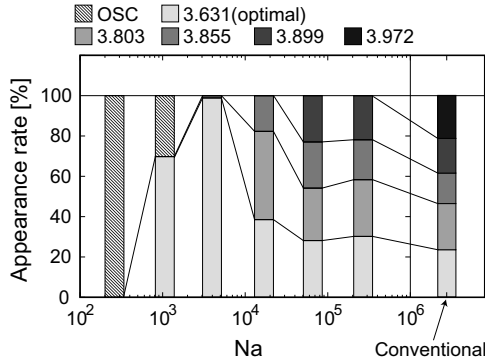
Fig. 3. Network state dependency on the accumulate time N_a. The parameter sets are $A = 5.08$, $C = 0.10$, and $B = 1.59$ ($B/4C = 3.98$). The rightmost bar shows the result in conventional network.

5   Conclusion

The network with the higher order connections is useful to solve the combinatorial optimization problems. However, the network topology is so complicated that implementation on hardware is not easy. We introduce the stochastic logic architecture to the discrete hysteresis network with the higher order connections and show the proposed network can present the solutions accordingly.

It is the future work to implement the network with the higher order connections by using the stochastic logic based on the result shown in this paper.

References

1. Hayakawa, Y., Nakajima, K.: Design of the inverse function delayed neural network for solving combinatorial optimization problems. IEEE Trans. Neural Netw. 21(2), 224–237 (2010)
2. Sota, T., Hayakawa, Y., Sato, S., Nakajima, K.: An application of higher order connection to inverse function delayed network. Nonlinear Theory and Its Applications, IEICE 2(2), 180–197 (2011)
3. Kondo, Y., Sawada, Y.: Functional abilities of a stochastic logic nerual network. IEEE Trans. Neural Netw. 3(3), 434–443 (1992)
4. Sota, T., Hayakawa, Y., Sato, S., Nakajima, K.: Discrete higher order inverse function delayed network. In: Proc. NOLTA 2010, pp. 615–618 (2010)

Dynamic Response Behaviors of a Generalized Asynchronous Digital Spiking Neuron Model

Takashi Matsubara and Hiroyuki Torikai

Department of Systems Innovation, Graduate School of Engineering Science, Osaka
University. 1-3 Machikaneyama-cho, Toyonaka, Osaka 560-8531, Japan
matubara@hopf.sys.es.osaka-u.ac.jp, torikai@sys.es.osaka-u.ac.jp
http://ushiolab.sys.es.osaka-u.ac.jp/

Abstract. A generalized asynchronous digital spiking neuron model
that can be implemented by an asynchronous sequential logic circuit
is presented. The presented model is the most generalized version of
asynchronous sequential logic circuit based neurons, where the sensitiv-
ity of its vector field to a stimulation input is generalized. It is clarified
that, the generalization enables the model to exhibit various nonlinear
responses characteristics that is classified into four groups. In addition,
it is clarified that the generalization enables the model to exhibit typi-
cal dynamic response behaviors having prominent features observed in
biological and model neurons.

Keywords: Neuron model, Sequential logic circuit, Cellular automaton,
Nonlinear dynamics.

1 Introduction

Various spiking neuron models suited for electronic circuit implementations have
been proposed so far, where there exist two major approaches: (i) an analog ap-
proach that implements a nonlinear *ordinary differential equation* (ab. ODE) in
an analog nonlinear circuit [1,2,3,4], and (ii) a digital approach that implements
a numerical integration in a digital processor [5,6,7]. Recently, an alternative
hardware-oriented neuron modeling approach has been proposed, where a non-
linear dynamics of a neuron is modeled by an asynchronous cellular automaton
that is implemented by an asynchronous sequential logic circuit [8,9,10,11]. In
this paper, a *generalized asynchronous digital spiking neuron model* (ab. GDN)
is presented, where the sensitivity of its vector field to a stimulation input is
generalized. As illustrated in Fig. 1, the GDN consists of registers, logic gates,
and reconfigurable wires, where the pattern of the wires is a control parame-
ter that determines the nonlinear dynamics of the GDN. It is clarified that the
generalization enables the GDN to realize the following novel nonlinear response
characteristics and dynamic response behaviors. (1) According to [12], nonlinear
response characteristics of biological and model neurons can be classified into
four groups depending on the two features: existence of bistability and existence
of subthreshold oscillation. The GDN can exhibit these features and reproduce
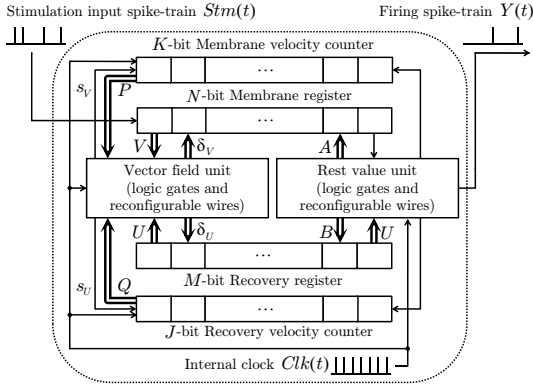
Fig. 1. Generalized asynchronous digital spiking neuron model (ab. GDN)

all the four groups of nonlinear response characteristics thanks to the generalization in this paper. (2) According to [13,12], biological and model neurons exhibit dynamic response behaviors having prominent features that can be classified into fifteen excitatory types and five inhibitory types. The GDN can exhibit all the five types of inhibitory dynamic response behaviors thanks to the generalization in this paper, whereas the previous asynchronous sequential logic circuit based neurons can exhibit only excitatory responses [8,9,10,11].

2 Generalized Asynchronous Digital Spiking Neuron

In this section, a *generalized asynchronous digital spiking neuron model* (ab. GDN), whose diagram is shown in Fig. 1, is presented. The GDN has the following four registers. (1) The *membrane register* is an N-bit bi-directional shift register having an integer state $V \in \mathbf{Z}_N \equiv \{0, \cdots, N-1\}$ by using the one-hot coding manner, where "$\equiv$" denotes "is defined by". From a neuron model viewpoint, the state V can be regarded as a *membrane potential*. (2) The *recovery register* is an M-bit bi-directional shift register having an integer state $U \in \mathbf{Z}_M \equiv \{0, \cdots, M-1\}$ by using the one-hot coding manner. From a neuron model viewpoint, the state U can be regarded as a *recovery variable*. (3) The *membrane velocity counter* is a K-bit register having an integer state $P \in \mathbf{Z}_K \equiv \{0, \cdots, K-1\}$ by using the thermometer coding manner. The state P controls a velocity of the membrane potential V. (4) The *recovery velocity counter* is a J-bit register having an integer state $Q \in \mathbf{Z}_J \equiv \{0, \cdots, J-1\}$ by using the thermometer coding manner. The state Q controls a velocity of the recovery variable U. The states V, U, P, and Q are clamped to the range $[0, N-1]$, $[0, M-1]$, $[0, K-1]$, and $[0, J-1]$, respectively. As shown in Fig. 1, the registers and the counters are connected to each other via the following two memoryless units. (i) The *vector field unit* consists of logic gates and reconfigurable wires. This unit determines the characteristics of a vector field of the states (V, U) as its name implies. (ii) The *reset value unit* consists of logic gates and reconfigurable wires. From a neuron model viewpoint, this unit determines
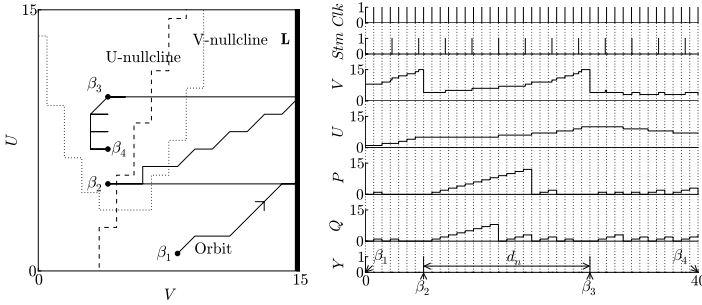
Fig. 2. A phase plane and state transitions. The nullclines are defined in (6). The bit lengths of the registers and the counters are $N = M = K = J = 16$. The parameters are $\boldsymbol{\Gamma} = (7, 0.3, 0.2, 3, 0.1, 16, 0.5, 0.3, 0)$ defined in (7). A periodic stimulation input spike-train $Stm(t)$ with a frequency 0.312 via the synaptic weight $W = 1$ is applied to the GDN.

values to which the states (V, U) are reset when the GDN fires, as its name implies. The GDN accepts a periodic *internal clock* $Clk(t)$ described by

$$Clk(t) = \begin{cases} 1 & \text{if } t \pmod 1 = 0, \\ 0 & \text{otherwise,} \end{cases}$$

where $t \in [0, \infty)$ is a continuous time. In the next subsection A, autonomous behaviors of the GDN (i.e., behaviors when no stimulation input spike-train $Stm(t)$ is applied) are investigated. After that, in the subsection B, non-autonomous behaviors of the GDN (i.e., behaviors when a stimulation input spike-train $Stm(t)$ is applied) are investigated.

2.1 Autonomous Behaviors

Let us begin with defining the following subset $\mathbf{L}$ in the state space $\mathbf{Z}_N \times \mathbf{Z}_M$ (see also Fig. 2).

$$\mathbf{L} \equiv \{(V, U) | V = N - 1, U \in \mathbf{Z}_M\} \subset \mathbf{Z}_N \times \mathbf{Z}_M.$$

From a neuron model viewpoint, $\mathbf{L}$ can be regarded as a *firing threshold*.

First, let us consider the case of $(V, U) \notin \mathbf{L}$. In this case, the reset value unit in Fig. 1 does not work and thus it is now excluded from the consideration. As shown in Fig. 1, the velocity counters accept the internal clock $Clk(t)$ and signals $(s_V, s_U) \in \{0, 1\}^2$ from the vector field unit. Then the internal clock $Clk(t)$ triggers transitions of the states (P, Q) of the velocity counters as follows.

$$P(t^+) = \begin{cases} P(t) + 1 & \text{if } s_V(t) = 0, Clk(t) = 1, \\ 0 & \text{if } s_V(t) = 1, Clk(t) = 1, \\ P(t) & \text{otherwise,} \end{cases}$$

$$Q(t^+) = \begin{cases} Q(t) + 1 & \text{if } s_U(t) = 0, Clk(t) = 1, \\ 0 & \text{if } s_U(t) = 1, Clk(t) = 1, \\ Q(t) & \text{otherwise,} \end{cases}$$

(1)

where t^+ denotes $\lim_{\epsilon \to +0}(t + \epsilon)$. As shown in Fig. 1, the two registers accept signals $(\delta_V, \delta_U) \in \{-1, 0, 1\}^2$ from the vector field unit. The signals (δ_V, δ_U) trigger transitions of the states (V, U) of the two registers as follows.

$$V(t^+) = V(t) + \delta_V(t), \quad U(t^+) = U(t) + \delta_U(t). \tag{2}$$

Inspired by Izhikevich's neuron model [14,12], the following design procedure of the signals $(s_V, s_U, \delta_V, \delta_U)$ is proposed that enable the GDN to exhibit various neuron-like nonlinear behaviors. The following *vector field functions* are prepared.

$$\mathcal{F}(V, U) = N(\gamma_1 (V/N - \gamma_2)^2 + \gamma_3 - U/M)/\lambda,$$
$$\mathcal{G}(V, U) = \mu M(\gamma_4 (V/N - \gamma_2) + (\gamma_3 + \gamma_5) - U/M)/\lambda,$$

where $(\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \lambda, \mu)$ are parameters. From a neuron model viewpoint, the functions $(\mathcal{F}(V, U), \mathcal{G}(V, U))$ can be regarded as *vector fields*, as their names imply. Using the vector field functions $(\mathcal{F}(V, U), \mathcal{G}(V, U))$, the signals (s_V, s_U) are generated as follows.

$$s_V = \begin{cases} 1 & \text{if } P \geq P_h(V, U), \\ 0 & \text{if otherwise,} \end{cases} \quad s_U = \begin{cases} 1 & \text{if } Q \geq Q_h(V, U), \\ 0 & \text{if otherwise,} \end{cases}$$

where the *velocity functions* (P_h, Q_h) are designed as follows.

$$P_h(V, U) = \lfloor |\mathcal{F}^{-1}(V, U)| \rfloor - 1, \quad Q_h(V, U) = \lfloor |\mathcal{G}^{-1}(V, U)| \rfloor - 1,$$

where the function $\lfloor x \rfloor$ gives the integer part of a real number x, and $P_h(V, U)$, $Q_h(V, U)$ are clamped to the range $[0, K-1]$, $[0, J-1]$, respectively. The signals (δ_V, δ_U) are generated as follows.

$$\delta_V = \begin{cases} \text{sgn}(\mathcal{F}(V, U)) & \text{if } P \geq P_h(V, U), Clk(t) = 1, \\ 0 & \text{otherwise,} \end{cases}$$

$$\delta_U = \begin{cases} \text{sgn}(\mathcal{G}(V, U)) & \text{if } Q \geq Q_h(V, U), Clk(t) = 1, \\ 0 & \text{otherwise.} \end{cases}$$

where the signum function $\text{sgn}(x)$ gives the sign of a real number x.

Second, let us consider the case of $(V, U) \in \mathbf{L}$. In this case, the reset value unit in Fig. 1 plays an important role in the dynamics and thus it is now focused on. As shown in Fig. 1, the reset value unit detects the situation $(V, U) \in \mathbf{L}$ and generates two integer signals $(A, B) \in \mathbf{Z}_N \times \mathbf{Z}_M$ by using the one-hot coding manners, where the signals (A, B) are called *reset values*. If the states (V, U) enter into the firing threshold $\mathbf{L}$, the following reset occurs.

$$(V(t^+), U(t^+), P(t^+), Q(t^+)) =$$
$$\begin{cases} (A, B, 0, 0) & \text{if } (V, U) \in \mathbf{L}, Clk(t) = 1, \\ (V(t), U(t), P(t), Q(t)) & \text{otherwise.} \end{cases} \tag{3}$$

Inspired by Izhikevich's neuron model [14,12], the following state-dependent reset values $(A, B(U))$ are presented that enable the GDN to exhibit various neuron-like nonlinear behaviors.

$$A = \lfloor \rho_1 N \rfloor, \quad B(U) = U + \lfloor \rho_2 M \rfloor,$$

where the reset value unit accepts the recovery variable U as shown in Fig. 1, (ρ_1, ρ_2) are parameters, and $A, B(U)$ are clamped to the range $[0, N-1]$, $[0, M-1]$, respectively. When the firing reset occurs, the GDN generates the following *firing spike-train* $Y(t)$.

$$Y(t) = \begin{cases} 1 & \text{if } (V(t), U(t)) \in \mathbf{L}, Clk(t) = 1, \\ 0 & \text{otherwise}. \end{cases} \quad (4)$$

2.2 Non-autonomous Behaviors

Let us now apply the following stimulation input spike-train $Stm(t)$ to the GDN.

$$Stm(t) = \begin{cases} W & \text{if } t = t_1, t_2, \cdots, \\ 0 & \text{otherwise}, \end{cases}$$

where $t = t_1, t_2, \cdots$ are input spike positions and $W \in \{-1, 1\}$ is a parameter. From a neuron model viewpoint, the stimulation input spike $Stm(t)$ can be regarded as a *stimulation input* and W can be regarded as a *synaptic weight*. A post-synaptic stimulation spike $Stm = W$ induces a transition of the membrane potential V as follows.

$$V(t^+) = \begin{cases} V(t) + W & Stm(t) = W, \\ V(t) & \text{otherwise}. \end{cases} \quad (5)$$

The stimulation input spike-train $Stm(t)$ accelerates and decelerates the increase velocity of the membrane potential V, depending on the sign of W. Fig. 2 shows basic non-autonomous behaviors of the GDN, where

$$\begin{aligned} \text{V-nullcline} &= \text{Border between } D_V \in \{-1, 0\} \text{ and } D_V = 1, \\ \text{U-nullcline} &= \text{Border between } D_U \in \{-1, 0\} \text{ and } D_U = 1. \end{aligned} \quad (6)$$

As a result, the dynamics of the GDN is described by (1)–(5), and is characterized by the following parameters.

$$\mathbf{\Gamma} = (\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \lambda, \mu, \rho_1, \rho_2). \quad (7)$$

3 Reproduction of Various Nonlinear Responses Characteristics and Dynamic Response Behaviors

In this section, it is shown that the GDN can exhibit various nonlinear responses characteristics and dynamic response behaviors of biological and model neurons.
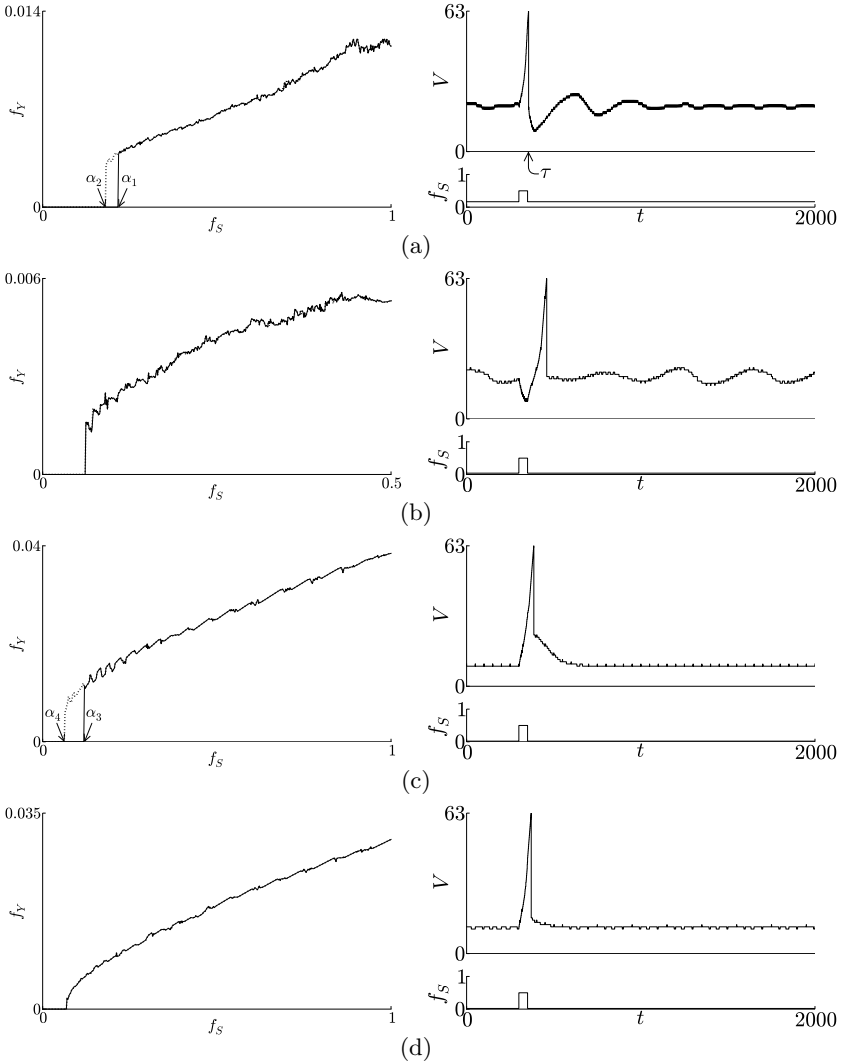
Fig. 3. Nonlinear response characteristics (left) and dynamic response behaviors (right) of the GDN. A solid line (dashed line) denotes the nonlinear response characteristics when the input frequency f_S is increased (decreased) from a small value (a large value). The bit lengths are $M = N = K = J = 64$. (a) The parameters are $\boldsymbol{\Gamma} = (7, 0.3, 0.2, 3, 0.1, 64, 0.5, 0.3, 0)$ and $W = 1$. Bistability is observed between the arrows α_1 and α_2 and subthreshold oscillation is observed. (b) The parameters are $\boldsymbol{\Gamma} = (7, 0.3, 0.5, -2.53, -0.05, 64, -0.33, 0.3, -0.04)$ and $W = -1$. No bistability is observed but subthreshold oscillation is observed. (c) The parameters are $\boldsymbol{\Gamma} = (7, 0.3, 0.2, -0.5, 0.1, 64, 4, 0.37, 0.35)$ and $W = 1$. Bistability is observed between the arrows α_3 and α_4 but no subthreshold oscillation is observed. (d) The parameters are $\boldsymbol{\Gamma} = (7, 0.3, 0.2, -0.5, 0.05, 64, 4, 0.25, 0.4)$ and $W = 1$. No bistability is observed and no subthreshold oscillation is observed.

For simplicity, we focus on the following periodic stimulation input spike-train $Stm(t)$.

$$Stm(t) = \begin{cases} W & \text{if } (t + \theta_0) \pmod{f_S^{-1}} = 0, \\ 0 & \text{otherwise,} \end{cases}$$

where f_S is an input frequency and $\theta_0 \in [0, f_S^{-1})$ is an initial input phase.

3.1   Groups of Nonlinear Response Characteristics

Let us begin with defining the following average firing frequency of the GDN.

$$f_Y = \left(\lim_{N \to \infty} \frac{1}{N} \sum_{n=1}^{N} d_n \right)^{-1},$$

where d_n is the n-th inter-spike interval as explained in Fig. 2, $f_Y = 0$ implies a resting state (i.e., no firing spike-train $Y(t)$ is generated), and $f_Y > 0$ implies a spiking state (i.e., a firing spike-train $Y(t)$ is generated). The GDN exhibits various nonlinear response characteristics (i.e., relations between the stimulation input f_S and the average firing frequency f_Y) and dynamic response behaviors (i.e., waveforms of the membrane potential V). Concerning them, the following terms are introduced.

Bistability: In the left figure of Fig. 3(a), when the input frequency f_S is increased from a small value (see the solid line), the average firing frequency f_Y jumps from zero (corresponding to a resting state) to a positive value (corresponding to a spiking state) at the arrow α_1. When the input frequency f_S is decreased from a large value (see the dashed line), the average firing frequency f_Y jumps from a positive value to zero at the arrow α_2. Then the average firing frequencies $f_Y = 0$ and $f_Y > 0$ overlap between the arrows α_1 and α_2. This overlap corresponds to a co-existence of a resting state ($f_Y = 0$) and a spiking state ($f_Y > 0$), where the GDN exhibits one of the co-existing states depending on the initial state. This type of co-existing phenomenon is called the *bistability* and is observed in biological and model neurons [13,12]. Bistability is also observed in the left figure of Fig. 3(c), and is not observed in the left figures of Figs. 3(b) and (d).

Subthreshold Oscillation: In the right figure of Figs. 3(a), the state V is oscillating without no firing spike $Y = 1$ generation after the GDN generates a firing spike $Y = 1$ at the arrow τ. This dynamic response behavior is called the *subthreshold oscillation* and is observed in biological and model neurons [13,12]. Subthreshold oscillation is also observed in the right figure of Fig. 3(b), and is not observed in the right figures of Figs. 3(c) and (d).

Based on the two features (i.e., existence of bistability and existence of subthreshold oscillation), the nonlinear response characteristics of the GDN are classified into four groups as follows.

		Existence of subthreshold oscillation	
		YES	NO
Existence of Bistability	YES	Fig. 3(a) (subcritical Hopf bif.)	Fig. 3(c) (saddle-node of invariant circle bif.)
	NO	Fig. 3(b) (supercritical Hopf bif.)	Fig. 3(d) (saddle-node on invariant circle bif.)

As shown in the above table, the GDN can reproduce all the possible four types of nonlinear response characteristics and related bifurcations, where these nonlinear response characteristics are observed also in biological and model neurons, and fundamental relations between the nonlinear response characteristics and the bifurcations are investigated in [12].

3.2   Inhibition Induced Dynamic Response Behaviors

According to [13,12], biological and model neurons typically exhibit dynamic response behaviors that can be classified into fifteen excitatory types and five inhibitory types as shown in Fig. 4. It has been shown so far that the asynchronous digital spiking neuron models [11] can exhibit all the fifteen excitatory dynamic response behaviors. Thanks to the generalization in this paper, the GDN can reproduce all the remaining five inhibitory dynamic response behaviors as shown in Figs. 4 and 5.

Inhibitory dynamic response behavior		
(a) Rebound spike	(b) Rebound bursting	(c) Threshold variability
(d) Inhibition-induced spiking	(e) Inhibition-induced bursting	

Excitatory dynamic response behavior		
(f) Tonic spiking	(g) Phasic spiking	(h) Tonic bursting
(i) Phasic bursting	(j) Mixed mode	(k) Spike frequency adaptation
(l) Class 1 excitable	(m) Class 2 excitable	(n) Spike latency
(o) Subthreshold oscillation	(p) Resonator	(q) Integrator
(r) Bistability	(s) Depolarizing after-potential	(t) Accommodation

Neuron model	Inhibitory					Excitatory dynamic response behavior														
	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)	(l)	(m)	(n)	(o)	(p)	(q)	(r)	(s)	(t)
Izhikevich [12]	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
Hodgkin Huxley [15]	+	+	+	+	-	+	+	+	-	-	+	+	+	+	+	+	+	+	+	+
RDN [10]	-	-	-	-	-	+	·	+	·	+	+	·	+	+	·	+	+	·	+	-
ADN [11]	-	-	-	-	-	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
GDN [this paper]	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

Fig. 4. Five inhibitory dynamic response behaviors and fifteen excitatory dynamic response behaviors. The lowest table summarizes reproduction abilities of dynamic response behaviors by typical neuron models and our models, where "+" denotes "reproducible", "-" denotes "not reproducible", and "·" denotes "partially reproducible". Each empty square denotes that sufficient parameter and initial value conditions are unknown but the model satisfies necessary conditions in principle [13].

(a) rebound spike   (b) rebound burst   (c) threshold variability

(d) inhibition induced spiking   (e) inhibition induced bursting
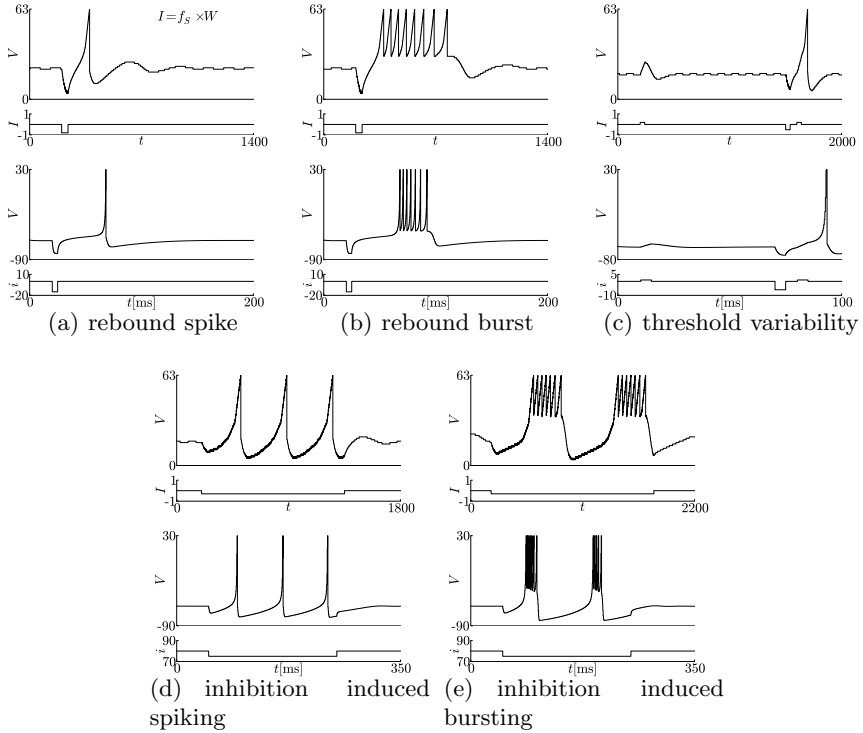
Fig. 5. Dynamic response behaviors of the GDN (each upper figure), where the *post-synaptic stimulation* to the GDN is defined $I = f_S \times W$, and the Izhikevich's neuron model in [13] (each lower figure). The bit lengths of the GDN are $M = N = K = J = 64$. The parameters $\boldsymbol{\Gamma}$ of the GDN and the heights of the post-synaptic stimulation I are as the followings. (a)$\boldsymbol{\Gamma} = (7, 0.3, 0.2, 3, -0.1, 64, 0.5, 0.3, 0)$, $I = -0.8$. (b)$\boldsymbol{\Gamma} = (7, 0.3, 0.2, 3, -0.1, 64, 0.5, 0.48, -0.42)$, $I = -0.8$. (c)$\boldsymbol{\Gamma} = (7, 0.3, 0.2, 3, 0.1, 64, 0.5, 0.3, 0)$, $I = 0.2, -0.5$. (d)$\boldsymbol{\Gamma} = (7, 0.3, 0.5, -5, -0.2, 64, -0.2, 0.3, 0)$, $I = -0.3$. (e)$\boldsymbol{\Gamma} = (7, 0.3, 0.5, -5, 0, 64, -0.1, 0.55, -0.1)$, $I = -0.3$. The parameter values of Izhikevich's neuron model can be found in [13].

4 Conclusion

The *generalized asynchronous digital spiking neuron model* (ab. GDN) whose dynamics is described by the asynchronous cellular automaton is proposed, where the sensitivity of its vector field to the stimulation input is generalized. It has been shown that, thanks to the generalization in this paper, the GDN can exhibit the combinations of the two features: existence of bistability and existence of sub-threshold oscillation that classify nonlinear response characteristics into the four groups. It has been also shown that the GDN can exhibit all the five types of inhibitory dynamic response behaviors. Thus the GDN can exhibit all the twenty types of dynamic response behaviors having prominent features. These properties will be keys to develop applications of the GDN such as neural prosthesis chip

and artificial pulsed neural network processor. Future problems include: clarification of relationships between the parameters of the GDN and experimentally measurable parameters of biological neurons, development of an on-chip learning algorithm of the GDN, and development of a neuroscience-aware network of GDNs. The authors would like to thank Professor Toshimitsu Ushio of Osaka University for valuable discussions. This work is partially supported by the Center of Excellence for Founding Ambient Information Society Infrastructure, Osaka University, Japan, and KAKENHI (21700253).

References

1. Arthur, J.V., Boahen, K.A.: Silicon-Neuron Design: A Dynamical Systems Approach. IEEE Trans. on Circuits and Systems I 58, 1034–1043 (2011)
2. Wijekoon, J.H.B., Dudek, P.: Compact silicon neuron circuit with spiking and bursting behaviour. Neural Networks 21, 524–534 (2008)
3. Vogelstein, R.J., Mallik, U., Vogelstein, J.T., Cauwenberghs, G.: Dynamically reconfigurable silicon array of spiking neurons with conductance-based synapses. IEEE Trans. on Neural Networks 18, 253–265 (2007)
4. Chen, H., Saïandghi, S., Buhry, L., Renaud, S.: Real-Time Simulation of Biologically Realistic Stochastic Neurons in VLSI. IEEE Trans. on Neural Networks 21, 1511–1517 (2010)
5. Weinstein, R.K., Reid, M.S., Lee, R.H.: Methodology and Design Flow for Assisted Neural-Model Implementations in FPGAs. IEEE Trans. on Neural Systems and Rehabilitation Engineering 15, 83–93 (2007)
6. Schoenauer, T., Atasoy, S., Mehrtash, N., Klar, H.: Neuropipe-chip: A digital neuro-processor for spiking neural networks. IEEE Trans. on Neural Networks 13, 205–213 (2002)
7. Moore, J.W., Ramon, F.: On numerical integration of the hodgkin and huxley equations for a membrane action potential. Journal of Theoretical Biology 45, 249–273 (1974)
8. Torikai, H., Hamanaka, H., Saito, T.: Reconfigurable digital spiking neuron and its pulse-coupled network: Basic characteristics and potential applications. IEEE Trans. on Circuits and Systems II 53, 734–738 (2006)
9. Hashimoto, S., Torikai, H.: A novel hybrid spiking neuron: Bifurcations, responses, and on-chip learning. IEEE Trans. on Circuits and Systems I 57, 2168–2181 (2010)
10. Hishiki, T., Torikai, H.: A Novel Rotate-and-Fire Digital Spiking Neuron And its Neuron-like Bifurcations and Responses. IEEE Trans. on Neural Networks 22, 752–767 (2011)
11. Matsubara, T., Torikai, H.: A novel asynchronous digital spiking neuron model and its various neuron-like bifurcations and responses. In: International Joint Conference on Neural Networks. IEEE Press, San Jose (accepted)
12. Izhikevich, E.M.: Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting. The MIT Press (2006)
13. Izhikevich, E.M.: Which model to use for cortical spiking neurons? IEEE Trans. on Neural Networks 15, 1063–1070 (2004)
14. Izhikevich, E.M.: Simple model of spiking neurons. IEEE Trans. on Neural Networks 14, 1569–1572 (2003)
15. Hodgkin, A.L., Huxley, A.F.: A quantitative description of membrane current and its application to conduction and excitation in nerve. The Journal of Physiology 117, 500–544 (1952)

Generalized PWC Analog Spiking Neuron Model and Reproduction of Fundamental Neurocomputational Properties

Yutaro Yamashita and Hiroyuki Torikai

Graduate School of Engineering Science, Osaka University,
1–3 Machikaneyama, Toyonaka, Osaka, Japan
yutaro@hopf.sys.es.osaka-u.ac.jp, torikai@sys.es.osaka-u.ac.jp

Abstract. An artificial spiking neuron model which has a generalized *piece-wise constant* (ab. PWC) vector field and state-dependent reset is proposed. Advantages of the PWC vector field include simplicity for hardware implementation, easiness to tune parameters, suitability for theoretical analysis based on theories on discontinuous *ordinary differential equations* (ab. ODEs). Using the analysis techniques of discontinuous ODEs, it is shown that the model can reproduce 6 types of the typical neuron-like responses (neurocomputational properties), the occurrence mechanisms of which have qualitative similarities to those of Izhikevich's simple neuron model.

Keywords: Spiking neuron model, Neuron-like responses, Bifurcation, Piece-wise constant vector field, VLSI neuron.

1 Introduction

Neurons exhibit various responses depending on stimulation inputs and parameter values. According to [1,2], 20 types of typical responses of neurons are called *the most fundamental neuroccomputational properties*. Many mathematical models (e.g., Izhikevich's simple neuronmodel [1,2,3] and Hodgkin-Huxlay's model [4]) have been studied intensively, where Izhikevich's simple neuron model can reproduce all the 20 types of typical responses of neurons. However, since typical control parameters of these mathematical neuron models are nonlinearities of *ordinary differential equations* (ab. ODEs), straightforward analog circuit implementations of these models [5,6,7,8,9,10,11,12] are sometimes cumbersome. Hence, as a hardware-oriented neuron model, we have proposed a *piece-wise constant* (ab. PWC) analog spiking neuron model which can be implemented by a simple electronic circuit [13]. The dynamics of the model is described by an ODE with PWC characteristics together with a state-dependent reset. It has been shown that the PWC analog spiking neuron model can reproduce a variety of excitatory responses of neurons [13,14]. In this paper, we propose a generalized PWC analog spiking neuron model. Thanks to the generalization, the model in this paper can reproduce typical neuron-like responses (i.e., *mixed*

mode, *spike frequency adaptation*, *spike latency*, *rebound spike*, *rebound burst*, *depolarizing after-potential*) that cannot be reproduced by the original model, where occurrence mechanisms of these responses have qualitative similarities to those of Izhikevich's simple neuron model [1,2,3]. Significances of this paper include the following points. **(1)** Advantages of the PWC vector field include: easy to implement by a compact electronic circuit, easy to tune parameter values, and suitability for theoretical analysis based on theories on discontinuous ODEs [15]. **(2)** The neural prosthesis is a recent hot topic, where a typical approach is to prosthesize a damaged part of neural systems by a digital processor [16,17]. On the other hand, sensory neurons should be prosthesized by analog electronic circuits since sensory neurons accept analog signals and it is not so efficient to utilize digital processor neurons together with analog-to-digital converters to implement them. Due to the advantages in the previous point (1), the proposed model will be a good (compact and tunable) candidate for a sensory neuron prosthesis as well as a hardware pulse-coupled neural network. **(3)** The proposed model can be regarded as a generalized version of a PWC oscillator in [18]-[20]. However, the oscillator is designed as an abstract chaotic oscillator and cannot exhibit neuron-like responses.

2   Piecewise Constant Analog Spiking Neuron Model

A generalized version of a *piece-wise constant* (ab. PWC) analog spiking neuron model [13] is proposed in Fig.1(a). The model consists of two capacitors, two *operational transconductance amplifiers* (ab. OTAs), a comparator, a monostable multivibrator, an analog switch, an amplifier, an adder, and an absolute value circuit. Fig.1(b) shows the characteristics of the OTA: it outputs a positive (negative) current if the differential voltage $v_\varepsilon = v^+ - v^-$ is positive (negative). From a viewpoint of neuron model, the capacitor voltages v and u can be regarded as an membrane potential and a recovery variable, respectively, as explained in the table in Fig.1. Also, an input voltage V_{in} and a constant voltage V_T can be regarded as a stimulation input and a spiking threshold, respectively. The constant voltage V_T is also regarded as a spike cut-off level [1]. If the membrane potential v reaches the spiking threshold V_T, the comparator (COMP) triggers the monostable multivibrator (MM) to generate a spike $Y = E$. The spike $Y = E$ closes the analog switch S for a short time, and then the membrane potential v is reset to a constant value V_B which is called a reset base. From a viewpoint of neuron model, the spike $Y = E$ is regarded as a firing spike or an action potential as explained in the table in Fig.1. The dynamics of the PWC analog spiking neuron model is described by the following equation.

$$
\begin{cases}
C\dot{v} = I_v(|v| + V_{in} - u) \\
C\dot{u} = I_u(av - u)
\end{cases}
\quad \text{if } v < V_T,
$$

$$
v(t^+) = V_B \qquad \text{if } v(t) = V_T,
$$

$$
I_v(v_\varepsilon) =
\begin{cases}
I_v^+ & \text{if } v_\varepsilon > 0 \\
-I_v^- & \text{if } v_\varepsilon < 0
\end{cases}
\tag{1}
$$
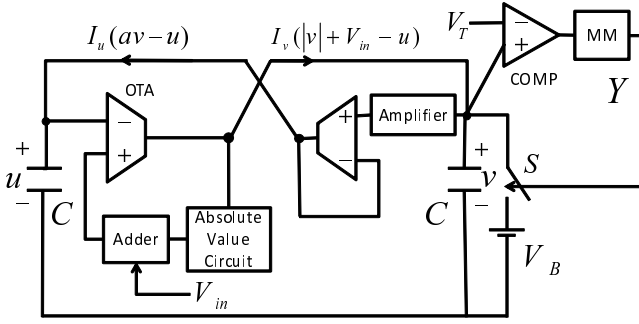
(a)
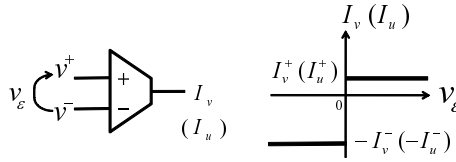
(b)

PWC spiking neuron	Meaning as a neuron model
Capacitor voltage v	Membrane potential
Capacitor voltage u	Recovery variable
Input voltage V_{in}	Stimulation input
Constant voltage V_T	Spiking threshold
Spike-train Y	Output firing spike-train

Fig. 1. PWC analog spiking neuron model. (a) Electrical circuit model. COMP and MM represent the comparator and the monostable multivibrator, respectively. (b) Characteristics of the *operational transconductance amplifier* (ab. OTA).

$$I_u(v_\varepsilon) = \begin{cases} I_u^+ & \text{if } v_\varepsilon > 0 \\ -I_u^- & \text{if } v_\varepsilon < 0 \end{cases}$$

$$Y(t^+) = \begin{cases} E & \text{if } v(t) = V_T \\ -E & \text{if } v(t) < V_T \end{cases}$$

where "˙" represents the time derivative, t^+ represents $\lim_{\varepsilon \to +0}(t + \varepsilon)$, I_v^+, I_v^-, I_u^+, $I_u^- > 0$ are assumed, and $v(0) \le V_T$ is assumed.

In the whole state space

$$\boldsymbol{S} \equiv \{(v, u)|v \le V_T\},$$

the following two borders are defined by the control voltages of the two OTAs (see also Fig. 2):

$$v\text{-nullcline} : \boldsymbol{\Sigma}_v \equiv \{(v, u)|u = |v| + V_{in}\},$$
$$u\text{-nullcline} : \boldsymbol{\Sigma}_u \equiv \{(v, u)|u = av\},$$

where " $\equiv$ " represents the "definition" hereafter. Since the borders play the same roles as nullclines of a smooth nonlinear ODE, the borders are called v-nullcline and u-nullcline. The nullclines divide the whole state space S into at most four subspaces having the following four vector fields:

$$(\dot{v}, \dot{u}) = \begin{cases} V^{++} \equiv (I_v^+/C, I_v^+/C) \text{ if } u < |v| + V_{in} \text{ and } u < av, \\ V^{-+} \equiv (I_v^-/C, I_v^+/C) \text{ if } u > |v| + V_{in} \text{ and } u < av, \\ V^{+-} \equiv (I_v^+/C, I_v^-/C) \text{ if } u < |v| + V_{in} \text{ and } u > av, \\ V^{--} \equiv (I_v^-/C, I_v^-/C) \text{ if } u > |v| + V_{in} \text{ and } u > av. \end{cases}$$

According to [13,15], the dynamics of the state (v, u) on the nullclines Σ_v and Σ_u can be categorized into *sliding mode* and *non-sliding mode* (we also say "*without sliding mode*"). If the mode is categorized into the sliding one, there exists some *sliding vector fields* on the nullclines Σ_v and Σ_u. More detailed explanations of the sliding mode dynamics of the PWC analog spiking neuron model [13] is omitted in this paper due to the page length limitation. Generalization of the model in this paper is due to relaxations of the parameter restrictions.

3   Analysis of Typical Neuron-Like Responses

In this section, we study six types of neuron-like responses of the generalized PWC analog spiking neuron model that cannnot be observed in the original model [13]. Fig.2 shows time waveforms and phase planes of the PWC analog spiking neuron model. In Fig.2,

$$v + Y' \tag{2}$$

is used to show neuron-like waveforms, i.e., spiking wave forms of $v + Y'$ are regarded as action potentials, where

$$Y'(t^+) = \begin{cases} K \text{ if } v(t) = V_T, \\ 0 \text{ if } v(t) < V_T, \end{cases} \tag{3}$$

and K is a parameter. Fig.3 shows time waveforms and phase planes of Izhikevich's simple neuron model described by the following equation.

$$\begin{cases} \dot{v} = 0.04v^2 + 5v + 140 - u + I \\ \dot{u} = a(bv - u) \end{cases} \tag{4}$$

$$\text{if } v \geq 30\text{mV}, \text{ then } \begin{cases} v \leftarrow c \\ u \leftarrow u + d \end{cases}$$

We make comparisons between our PWC analog spiking neuron model and Izhikevich's simple neuron model as the followings, where the parameters (C, V_T, K) of the PWC analog spiking neuron model are fixed to $(C, V_T, K) = (0.01, 1.0, 5.0)$.
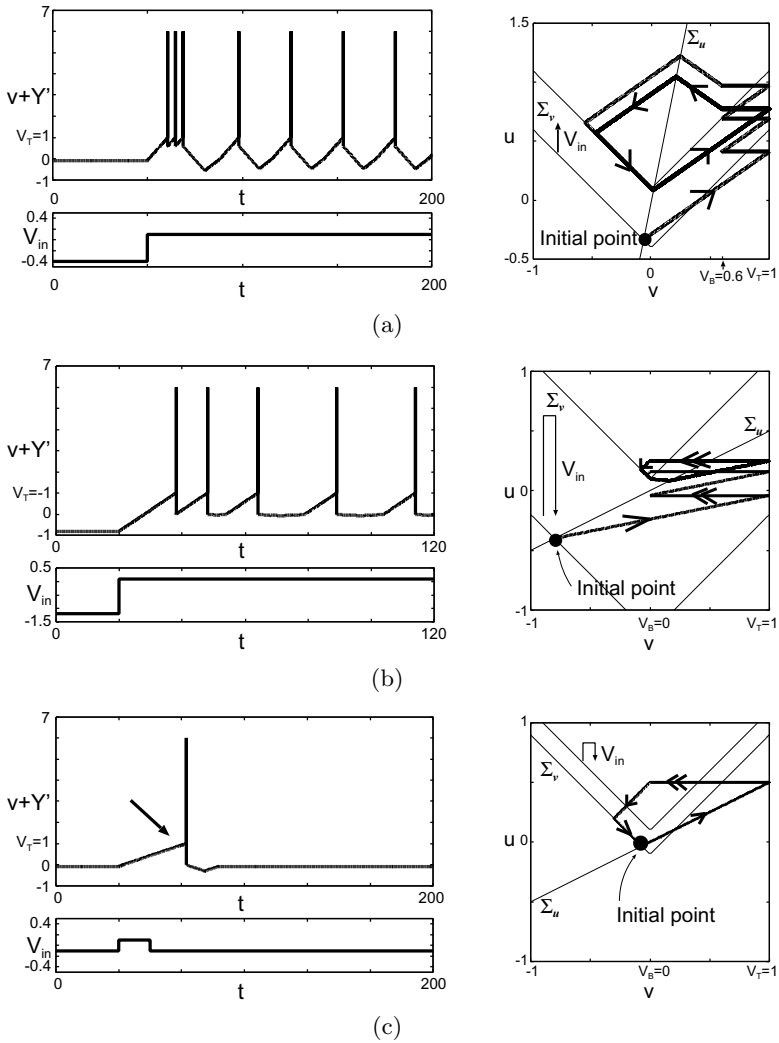
Fig. 2. Neuron-like responses of PWC analog spiking neuron model. (a) Mixed mode. The parameters are $a = 5.0$, $I_v^+ = 1.0$, $I_v^- = 1.0$, $I_u^+ = 0.7$, $I_u^- = 0.7$, $V_B = 0.6$. (b) Spike frequency adaptation. The parameters are $a = 0.5$, $I_v^+ = 1.0$, $I_v^- = 0.1$, $I_u^+ = 0.2$, $I_u^- = 0.1$, $V_B = 0.0$. (c) Spike latency. The parameters are $a = 0.5$, $I_v^+ = 0.1$, $I_v^- = 0.1$, $I_u^+ = 0.4$, $I_u^- = 0.1$, $V_B = 0.0$. (d) Rebound spike. The parameters are $a = 0.5$, $I_v^+ = 1.0$, $I_v^- = 0.1$, $I_u^+ = 0.1$, $I_u^- = 0.1$, $V_B = 0.0$. (e) Rebound burst. The parameters are $a = 5.0$, $I_v^+ = 1.0$, $I_v^- = 1.0$, $I_u^+ = 0.3$, $I_u^- = 0.3$, $V_B = 0.6$. (f) Depolarizing after-potential. The parameters are $a = -0.5$, $I_v^+ = 1.0$, $I_v^- = 0.5$, $I_u^+ = 0.21$, $I_u^- = 0.5$, $V_B = -0.5$.

(d)

(e)

(f)
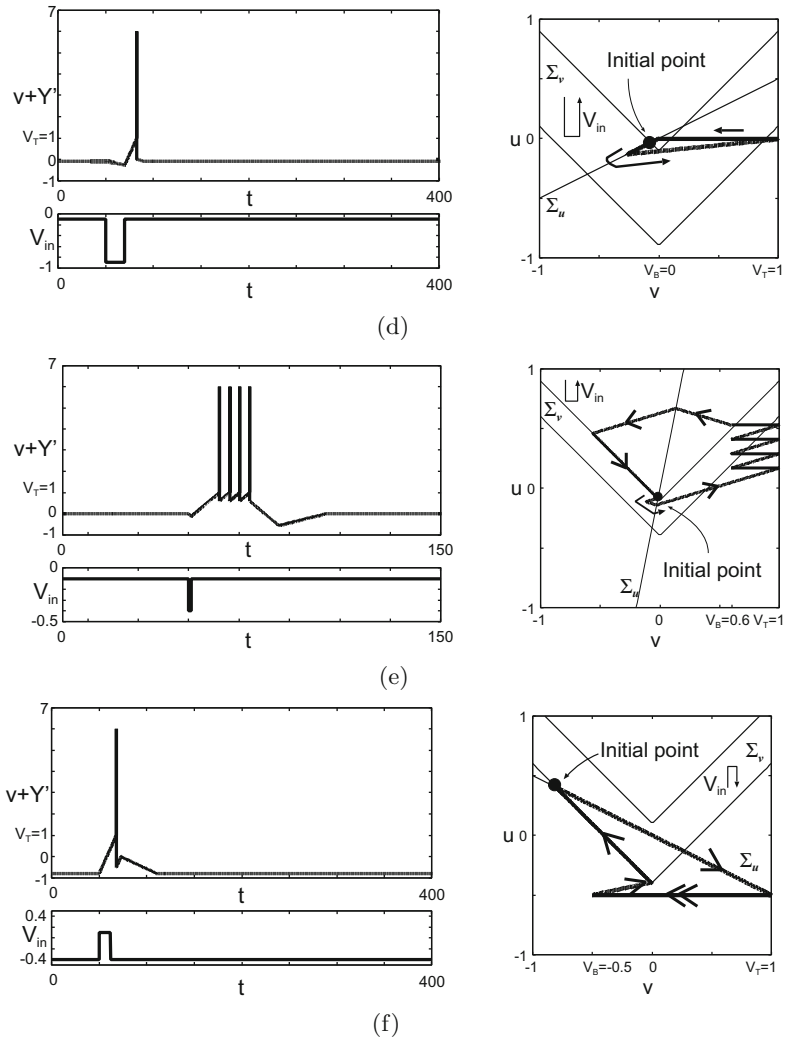
Fig. 2 (*Continued*)

A. Mixed Mode: In Fig. 2(a), there exists a resting state at first. Next the stimulation input V_{in} is increased and the stable equilibrium point of the resting state loses the stability by a *border-collision bifurcation* [15]. Then the PWC analog spikinge neuron model exhibits a *tonic spiking* [1,2] after generating bursting spikes (*phasic bursting* [1,2]). This type of response is called the *mixed mode* [1,2]. The above occurrence mechanism of the mixed mode is qualitatively similar to that of Izhikevich's simple neuron model shown in Fig. 3(a).
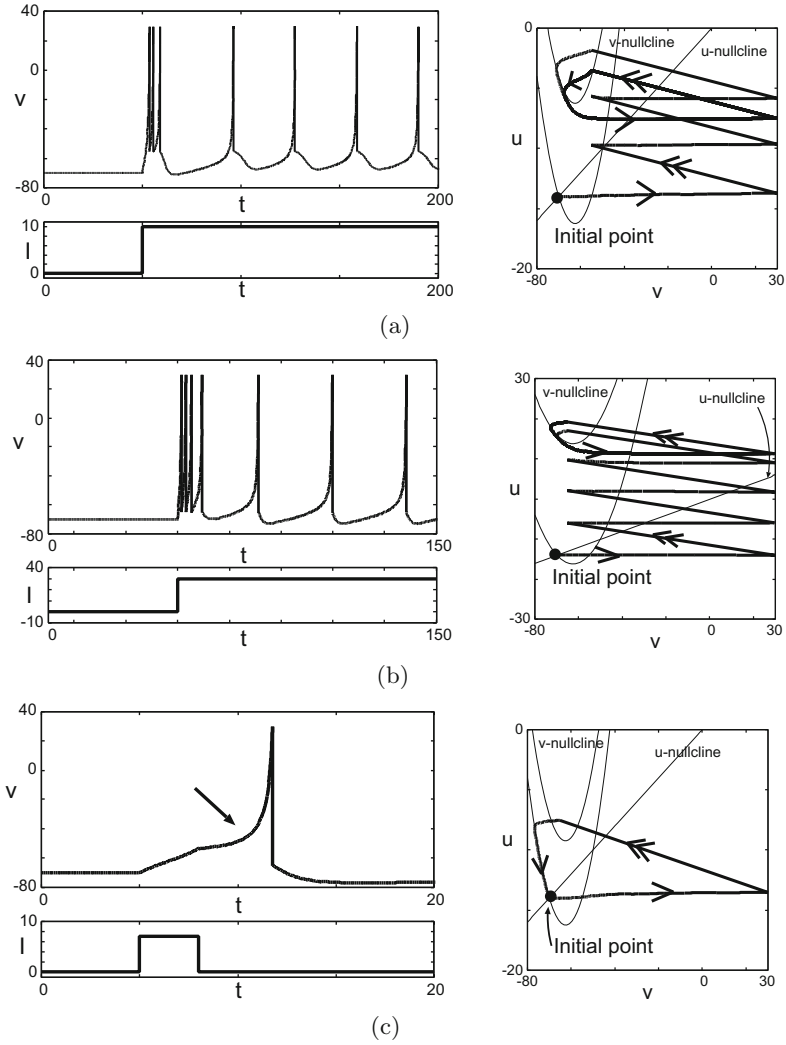
Fig. 3. Neuron-like responses of Izhikevich's simple neuron model. (a) Mixed mode. The parameters are $a = 0.02$, $b = 0.2$, $c = -55$, $d = 4$. (b) Spike frequency adaptation. The parameters are $a = 0.01$, $b = 0.2$, $c = -65$, $d = 8$. (c) Spike latency. The parameters are $a = 0.02$, $b = 0.2$, $c = -65$, $d = 6$. (d) Rebound spike. The parameters are $a = 0.03$, $b = 0.25$, $c = -60$, $d = 4$. (e) Rebound burst. The parameters are $a = 0.03$, $b = 0.25$, $c = -52$, $d = 0$. (f) Depolarizing after-potential. The parameters are $a =$, $b = 0.2$, $c = -60$, $d = -20$.

(d)

(e)

(f)
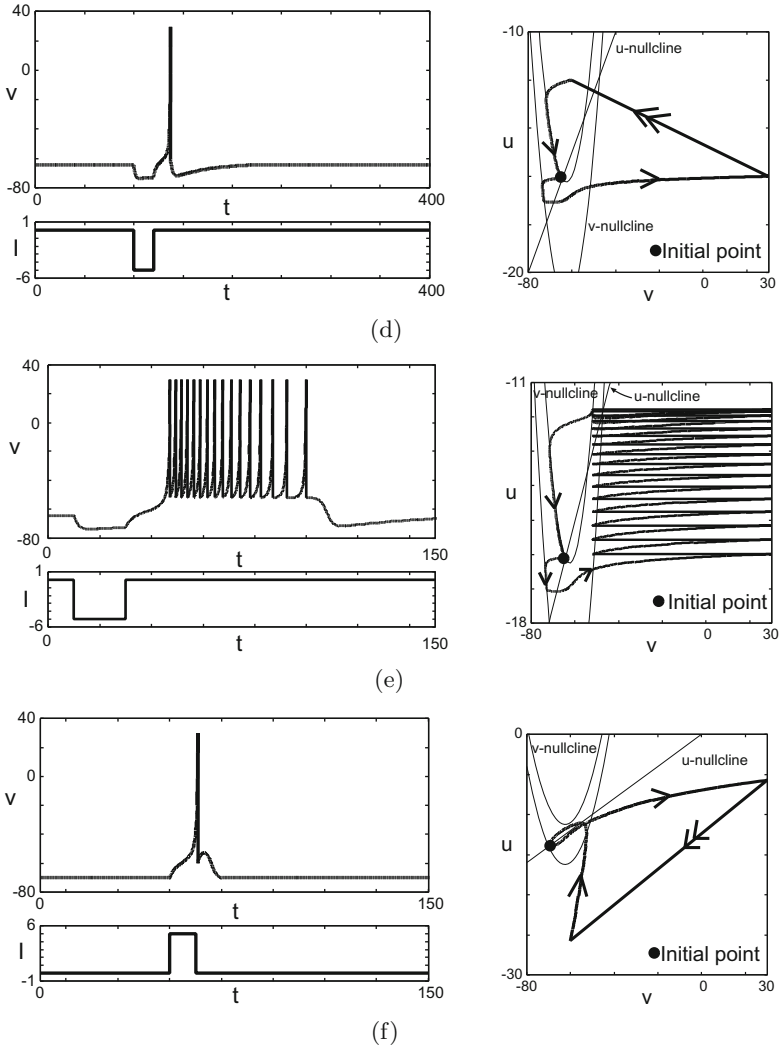
Fig. 3 (*Continued*)

B. Spike Frequency Adaptation : In Fig. 2(b), there exists a resting state at first. Next the stimulation input V_{in} is increased and the stable equilibrium point of the resting state loses the stability by a *border-collision bifurcation* [15] (we call this border-collision bifurcation as *saddle-node on invariant circle type border-collision bifurcation* [13] because the bifurcation has qualitative similarities to the *saddle-node on invariant circle bifurcation* [21]). Then the PWC analog spiking neuron model generates tonic spiking with a decreasing spike

frequency. This type of response is called the *spike frequency adaptation* [1,2]. The above occurrence mechanism of the spike frequency adaptation is qualitatively similar to that of Izhikevich's simple neuron model shown in Fig.3(b).

C. Spike Latency: In Fig.2(c), there exists a resting state at first. Next an excitatory pulse input V_{in} is injected and the stable equilibrium point of the resting state loses the stability by a saddle-node on invariant circle type border-collision bifurcation. Then the PWC analog spiking neuron model generates a spike that is delayed with respect to the onset of the stimulation input V_{in}. This type of response is called the *spike latency* [1,2]. The above occurrence mechanism of the spike latency is qualitatively similar to that of Izhikevich's simple neuron model shown in Fig.3(c).

D. Rebound Spike: In Fig.2(d), there exists a resting state at first. Next an inhibitory pulse input V_{in} is injected and the PWC analog spiking neuron model generates a single spike. This type of response is called the *rebound spike* [1,2]. The above occurrence mechanism of the rebound spike is qualitatively similar to that of Izhikevich's simple neuron model shown in Fig.3(d).

E. Rebound Burst: In Fig.2(e), there exists a resting state at first. Next an inhibitory pulse input V_{in} is injected and the PWC analog spiking neuron model generates bursting spikes. This type of response is called the *rebound burst* [1,2]. The above occurrence mechanism of the rebound burst is qualitatively similar to that of Izhikevich's simple neuron model shown in Fig.3(e).

F. Depolarizing After-Potential: In Fig.2(f), there exists a resting state at first. Next an excitatory pulse input V_{in} is injected and the stable equilibrium point of the resting state loses the stability by a saddle-node on invariant circle type border-collision bifurcation.1 Then PWC analog spiking neuron model generates a single spike and the model shows depolarized after-potential after that. This type of response is called the *depolarizing after-potential* [1,2]. The above occurrence mechanism of the depolarizing after-potential is qualitatively similar to that of Izhikevich's simple neuron model shown in Fig.3(f).

Table 1. Reproducabilities of typical neuron-like responses (i.e., neurocomputational properties [1,2]) by the PWC analog spiking neuron model

Mixed mode		Tonic spiking	
Spiking frequency adaptation		Tonic bursting	See [13].
Spike latency	This paper.	Bistability	
Rebound spike	(*)	Class 1 excitable	
Rebound burst		Resonator	
depolarizing after-potential		Integrator	
Phasic spiking		Threshold variability	Unclear so far.
Phasic bursting	See [14].	Accommodation	
Class 2 excitable		Inhibition-induced spiking	
Subthreshold oscillation		Inhibition-induced bursting	

(*) The six neuron-like responses can be reproduced due to the generalization in this paper.

The 20 Types of Neuron-Like Responses: The PWC analog spiking neuron model can reproduce a variety of responses as shown in Table 1. The responses explained as "unclear so far." in Table 1 will be investigated in a future paper.

4   Conclusions

We have proposed the generalized *piece-wise constant* (ab. PWC) analog spiking neuron model. It has been shown that , thanks to the generalization, the model can reproduce the typical neuron-like responses (i.e., *mixed mode*, *spike frequency adaptation*, *spike latency*, *rebound spike*, *rebound burst*, *depolarizing after-potential*), where the occurrence mechanisms of these neuron-like responses have qualitative similarities to those of Izhikevich's simple neuron model. Future problems include: (a) more in-depth theoretical analysis of responses of the model (i.e., reproduction of unclear neuron-like responses in Table 1), and (b) synthesis of a network of the PWC analog spiking neuron model and investigation of its applications.

References

1. Izhikevich, E.: Dynamical systems in neuroscience: The geometry of excitability and bursting. The MIT Press, Cambridge (2007)
2. Izhikevich, E.: Which model to use for cortical spiking neurons? IEEE Trans. Neural Netw. 15(5), 1063–1070 (2004)
3. Izhikevich, E.: Simple model of spiking neurons. IEEE Trans. Neural Netw. 14(6), 1569–1572 (2004)
4. Hodgkin, A., Huxley, A.: A quantitative description of membrane current and its application to conduction and excitation in nerve. The Journal of Physiology 117(4), 500–544 (1952)
5. Indiveri, G., et al.: A VLSI Array of Low-Power Spiking Neurons and Bistable Synapses With Spike-Timing Dependent Plasticity. IEEE Trans. Neural Netw. 17(1), 211–221 (2006)
6. Yu, T., Cauwenberghs, G.: Analog VLSI Biophysical Neurons and Synapses With Programmable Membrane Channel Kinetics. IEEE Trans. Biomed. Circuits Syst. 4(3), 139–148 (2010)
7. Asai, T., et al.: A subthreshold MOS neuron circuit based on the Volterra system. IEEE Trans. Neural Netw. 14(5), 1308–1312 (2003)
8. Farquhar, E., Hasler, P.: A bio-physically inspired silicon neuron. IEEE Trans. CAS-I 52(3), 477–488 (2005)
9. Basu, A., Hasler, P.: Nullcline-Based Design of a Silicon Neuron. IEEE Trans. CAS-I 57(11), 2938–2947 (2010)
10. Simoni, M., DeWeerth, S.: Adaptation in a VLSI model of a neuron. IEEE Trans. CAS-II 46(7), 967–970 (2002)

11. Rasche, C.: An a VLSI basis for dendritic adaptation. IEEE Trans. CAS-II 48(6), 600–605 (2001)
12. Kohno, T., Aihara, K.: A MOSFET-based model of a class 2 nerve membrane. IEEE Trans. Neural Netw. 16(3), 754–773 (2005)
13. Yamashita, Y., Torikai, H.: A Novel Piece-Wise Constant Analog Spiking Neuron Model and its Neuron-like Excitabilities. In: Proc. IEEE-INNS/IJCNN, pp. 717–724 (2011)
14. Yamashita, Y., Torikai, H.: Bifurcation-based synthesis of a PWC analog spiking neuron model. In: Proc. NOLTA (to appear, 2011)
15. Di Bernardo, M.: Piecewise-smooth dynamical systems: theory and applications. Springer, London (2008)
16. Berger, T., et al.: Restoring lost cognitive function. IEEE Eng. Med. Biol. Mag. 24(5), 30–44 (2005)
17. Berger, T., Glanzman, D.: Toward replacement parts for the brain: implantable biomimetic electronics as neural prostheses. The MIT Press, Cambridge (2005)
18. Matsuoka, Y., Hasegawa, T., Saito, T.: Chaotic Spike-Train with Line-Like Spectrum. IEICE Trans. Fund. E92-A(4), 1142–1147 (2009)
19. Matsuoka, Y., Saito, T.: A Simple Chaotic Spiking Oscillator Having Piecewise Constant Characteristics. IEICE Trans. Fund. E89-A(9), 2437–2440 (2006)
20. Tsubone, T., Saito, T.: Manifold Piecewise Constant Systems and Chaos. IEICE Trans. Fund. E82-A(8), 1619–1626 (1999)
21. Kuznetsov, Y.: Elements of applied bifurcation theory. Springer, New York (1998)

Implementation of Visual Attention System Using Artificial Retina Chip and Bottom-Up Saliency Map Model

Bumhwi Kim[1], Hirotsugu Okuno[2], Tetsuya Yagi[2], and Minho Lee[1,*]

[1] School of Electronical Engineering and, Kyungpook National University, Taegu, Korea
[2] Division of Electronic and Information Engineering, Osaka University, Osaka, Japan
bhkim@ee.knu.ac.kr, {h-okuno,yagi}@eei.eng.osaka-u.ac.jp,
mholee@knu.ac.kr

Abstract. This paper proposes a new hardware system for visual selective attention, in which a neuromorphic silicon retina chip is used as an input camera and a bottom-up saliency map model is implemented by a Field-Programmable Gate Array (FPGA) device. The proposed system mimics the roles of retina cells, V1 cells, and parts of lateral inferior parietal lobe (LIP), such as edge extraction, orientation, and selective attention response, respectively. The center surround difference and normalization for mimicking the roles of on-center and off-surround function in the lateral geniculate nucleus (LGN) are implemented by the FPGA. The integrated artificial retina chip with the FPGA successfully produces the human-like visual attention function, with small computational overhead. In order to apply this system to mobile robotic vision, the proposed system aims to low power dissipation and compactness. The experimental results show that the proposed system successfully generates the saliency information from natural scene.

Keywords: Visual selective attention, bottom-up saliency map, neuromorphic silicon retina.

1 Introduction

The human visual system can effortlessly detect an interesting area or an object within natural or cluttered scenes through a selective attention mechanism. This mechanism allows the human vision system to effectively process visual scenes with a higher level of complexity. The start point of the human visual system is the retina.

The retina can extract low level features like color, contrast and edge information. Those features are used to construct an input for selecting an interesting area in complex visual scenes through further processing. The silicon retina, as a neuromorphic device, can mimic the process of the human retina.

On the other hand, there have been several studies about the selective attention which process stimuli from the retina to the visual cortex. Itti, Koch, and Niebur

---

(1998) [1] introduced a brain-like model in order to generate a saliency map (SM). Koike and Saiki (2002) [2] proposed that a stochastic winner take all (WTA) enables the saliency-based search model to change search efficiency by varying the relative saliency, due to stochastic shifts of attention. Kadir and Brady (2001) [3] proposed an attention model integrating saliency, scale selection, and a content description, thus contrasting with many other approaches. Ramström and Christensen (2002) [4] calculated saliency with respect to a given task by using a multiscale pyramid and multiple cues. Their saliency computations were based on game theory concepts. And Jeong et al. [5] introduced a dynamic saliency model which considers temporal dynamics of saliency degrees changing through time at each salient point. This model is based on a modified static saliency model which additionally considers symmetry information. Ban et al. [6] proposed an affective saliency model which considers psychological distance as well as visual features.

Most of the selective attention models only utilized input images obtained from non-biological image sensor like a charge-coupled devices (CCD) or CMOS camera. In a recent paper, Indiveri (2008) [7] proposed a neuromorphic VLSI model for selective attention. Indiveri shows successful neuromorphic VLSI models of a selective attention system, which was applied to the visual tracking problem. They describe some examples of single-chip and multi-chip based selective attention system. However, the image size of the system is only 32 by 32, and it is too small to perceive important attentive features in natural input scenes.

Because our final goal is to make a mobile embedded system, the system needs to satisfy the following requirements: real-time computation, compact hardware, and low power consumption. In the present study, we developed a new neuromorphic selective attention system which satisfies the requirements by taking advantages of parallel computation of a field programmable gate array (FPGA) and a silicon retina which has 128 by 128 pixels and analog resistive networks.

In Section 2, we present the proposed model in detail. Experimental results are described in Section 3. Discussion and conclusions follow in Section 4.

2   Bottom-Up Saliency Model

2.1 Selective Attention Model

In the human visual processing system, there are two ways to recognize an object. One is the bottom-up saliency process and the other is top-down selective attention. The bottom-up saliency map (SM) model localizes the salient points of a natural scene based on data-driven processing. The top-down selective attention model which is concept-driven processing model based on pre-learnt information, performs a specific task such as object detection and recognition, preferable attention, and so forth. To make human-like selective attention model, both of the bottom-up SM and the top-down selective attention models are essentially needed. In our research, we want to implement a specific neuromorphic hardware system for mimicking human-like selective attention function in a vision system. As the first step of the hardware implementation, we consider the bottom-up SM model. Like the human visual processing system, we utilize the intensity and the edge features as a low level inputs.

Then we use the center surround difference and normalization (CSDN) module which is based on the function of the lateral geniculate nucleus (LGN), to generate the conspicuity map using the on-set and off-surround mechanism. And those conspicuity maps make a bottom-up SM by the feature integration theory which occurs in laterial intral-parietal cortex (LIP) of human brain [8].

2.2   Simplified Bottom-Up Saliency Map Model

For the compactness of the proposed system, we propose the simplified the bottom-up SM model for hardware implementation. The simplified bottom-up SM model utilizes intensity and edge information to generate the final saliency map. Figure 1 shows a simplified bottom-up saliency map model.
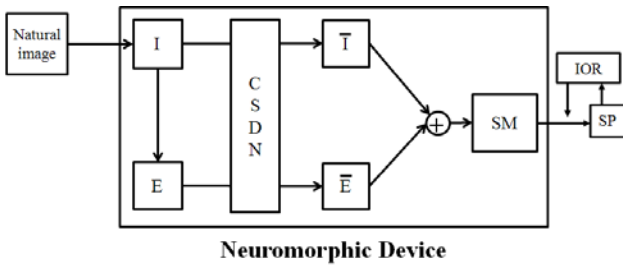
Fig. 1. The simplified bottom-up saliency map model for hardware implementation using silicon retina chip

From an input image, edge information has been extracted by applying a 3 by 3 Sobel edge operator. Then, intensity (I) and edge (E) information are transferred to the (CSDN) module. The intensity and edge conspicuity maps ($\overline{I}$ and $\overline{E}$) are calculated by accumulating and normalizing each feature map. Finally, the SM is extracted by accumulating two conspicuity maps. Salient points are extracted by the inhibition of return (IOR) mechanism [8, 9].

The simplified SM model reflects the part of functions in the retina cells, the LGN, the LIP and visual cortex. Although the retina cells can process color opponency as well as extract edge and intensity information, in the proposed simplified bottom-up SM model, we do not consider color information because the obtained image from the silicon retina is a gray scale image. Therefore, the proposed model considers two feature bases such as the intensity (I) and the edge (E). Moreover, the proposed model considers the on-center and off-surround operation of the LGN and the ganglian cells by implementing the CSDN of Gaussian pyramid images with different scales from 0 to n-th level, whereby each level is made by the sub-sampling of the n-th powers of 2. This reflects the non-uniform distribution of the retina-topic structure. Then, the center-surround mechanism is implemented in the model as the difference operation between the fine and coarse scales of the Gaussian pyramid images [8]. Consequently, two conspicuity maps are obtained by the Eqs.(1) and (2).

$$I(c,s) = \left| I(c) \otimes I(s) \right|$$
$$E(c,s) = \left| E(c) \otimes E(s) \right| \tag{1}$$

where "$\otimes$" represents interpolation to the finer scale and point-by-point subtraction, c and s are indices of the finer scale and the coarse scale, respectively. Totally, 8 feature maps are computed because intensity and edge individually have 5 different scales (2 finer scales and 3 coarse scales) [8, 12]. Each feature map is combined into the corresponding conspicuity map as shown in Eq. (2) where $\overline{I}$ and $\overline{E}$ stands for intensity and edge conspicuity maps, respectively. These are obtained through across-scale addition [5].

$$\overline{I} = \sum_{c=0}^{1} \sum_{s=c+2}^{c+3} N(I(c,s))$$
$$\overline{E} = \sum_{c=0}^{1} \sum_{s=c+2}^{c+3} N(E(c,s)) \tag{2}$$

Consequently, two conspicuity maps can be obtained by the center-surround difference and normalization (CSDN) algorithm [10]. A SM is simply generated by the summation of those two conspicuity maps as shown in Eq. (3).

$$SM = \overline{I} + \overline{E} \tag{3}$$

The inhibition-of-return (IOR) process is applied by the brain for efficient information processing [8, 9]. The IOR function prevents repeated attention to on already-attended area, which is an efficient mechanism for effectively processing of complex visual scenes. To select the proper scale of the salient areas, an entropy maximization approach is considered, which is based on Kadir's approach [3]. After masking this IOR region, the SM finds the next salient point that excludes the previous salient object.

3 Hardware Implementation

3.1 Field-Programmable Gate Array (FPGA) Implementation

Figure 2 shows the overall architecture of the proposed model. We utilize the silicon retina and its resistive network. The silicon retina used here has 128 x 128 pixels, each of which is composed of an active pixel sensor (APS), resistive networks, and differential amplifiers (ASPECTUS-U, Neuralimage Co. Ltd). The resistive network of the silicon retina computes the multiple Gaussian filters efficiently and instantaneously and transfers it to the Field-Programmable Gate Array (FPGA, Xilinx X3CS400) block that is for implementing a bottom-up SM model. The FPGA receives multiple images with different Gaussian scales within a single frame because the

value of resistance *Rs* is controlled by the FPGA via a digital to analog (D/A) converter. An intensity pyramid block gets images with 5 different scales from fine to coarse. An edge pyramid block calculates edge information from each intensity pyramid image. The CSDN block computes a difference of center and surround contrast between coarse and fine images of the Gaussian pyramid image. Then, two conspicuity map blocks compute each conspicuity map by accumulating 4 feature maps that are the outputs of CSDN block. After that, a saliency map is computed by summation of the conspicuity maps in every location. Then the computed SM is transferred to a PC through the USB interface (I/F) block. Finally, a salient region is represented in the PC. The salient points are detected by the inhibition-of-return (IOR) function in the PC. The size of each salient area is determined by using the entropy maximization process of the saliency map [5].
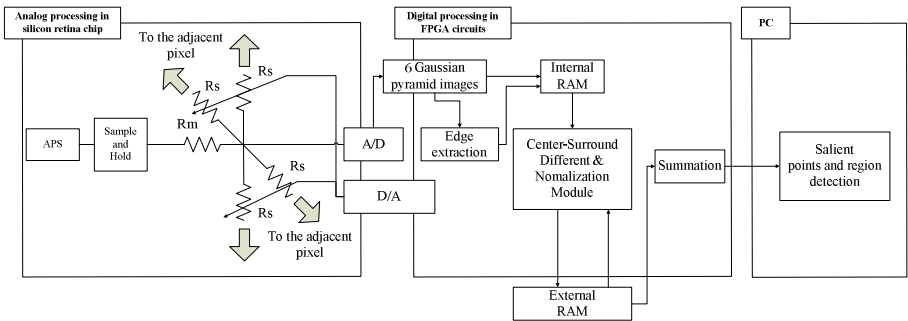
Fig. 2. The block diagram of our proposed visual selective attention system using a silicon retina chip and FPGA for bottom-up saliency map model

3.2   Modified CSDN Algorithm for FPGA

When we check the computational load for the saliency map processing, the CSDN block takes too much time. To solve this problem, we use an FPGA to calculate the CSDN process.

Due to small memory capacity of the FPGA block, the CSDN processing algorithm is modified in an effective way. The embedded hardware system including silicon retain chip contains two memories such as Block RAM (BRAM) and Static RAM (SRAM). The BRAM can store two 128 by 128 images and read/write the memory data simultaneously. And the SRAM can store 16 images. But, the SRAM only access memory one way at a time such as read or writes. While the BRAM has fast access to memory data, SRAM takes time to access its data. Considering the limitation of those memories, we modified the CSDN algorithm. Fig. 3 explains the modified CSDN algorithm for hardware implementation, which changes the input sequence so that the FPGA accesses two memory blocks with the smallest number of possible read and write.
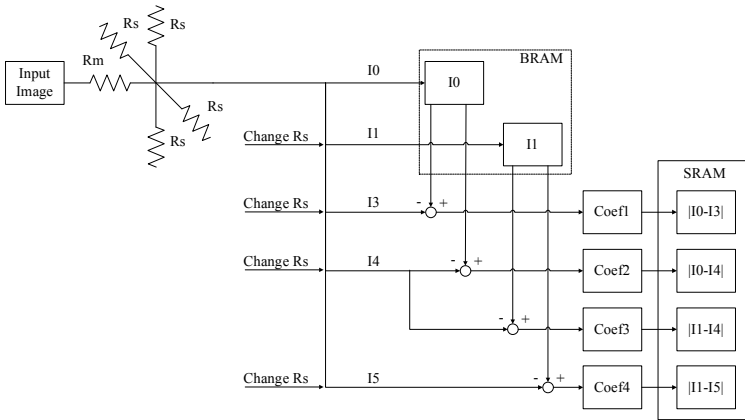
Fig. 3. The modified CSDN processing, I0-I5: pixels data of intensity (or edge) Gaussian pyramid image, Coef1-4: Coefficient value for data normalization, |*|: arithmetical absolute operation

3.3 Modified Normalization Algorithm for FPGA

In FPGA, dividing operations are computationally inefficient and block many gates. So we changed the normalization process as shown in Eq. (7)

$$N(i) = (i - \min_i) \times \frac{255}{\max_i - \min_i} \tag{7}$$

where $N(i)$ represents a normalized value of input i. And $\max_i$ and $\min_i$ represent maximum and minimum values of previous the frame, respectively. It takes additional computational load to find maximum and minimum values in Eq. (7). We utilize previous image's value because the silicon retina's frame rate is quite fast. In the FPGA, the value can be divided by 2's power value. When we use 2's power value for the difference between maximum and minimum values of intensity as a denominator, only 9 level result values are obtained. But, when we use an integer denominator for the difference between maximum and minimum values of intensity as a denominator in Eq. (7), we can obtain 30 levels of result values. To reduce division distortion, we programmed an integer division module. As a result, the division process converted into a multiplication process by utilizing Eq. (7). Then we can get a conspicuity map by summing 4 feature maps stored in SRAM. The saliency map is obtained by summation of intensity and edge conspicuity maps. Finally, the saliency map is transferred to PC via USB interface to extract saliency information.

4 Experiments

To evaluate our proposed model, we compared the bottom-up saliency results of the proposed system with those of the PC. Fig. 4 shows the example input images which
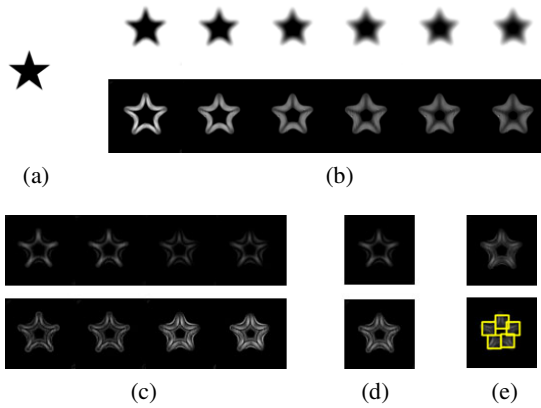
Fig. 4. Experimental results of proposed model, (a) original image, (b) Gaussian pyramid image (1st row: intensity, 2nd row: edge), (c) feature maps (1st row: intensity, 2nd row: edge), (d)Conspicuity maps (1st row: intensity, 2nd row: edge), (e) saliency map and its salient region
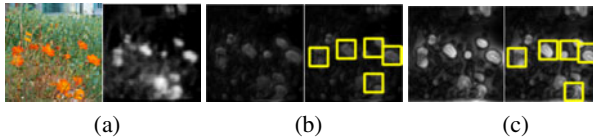
Fig. 5. Comparison of the results for selective attention process using PC and FPGA (a) original image and its captured image by silicon retina chip (b) results by FPGA, (c) results by PC

captures by silicon retina. And Fig. 5 shows the results using the FPGA and the PC, respectively.

To compare the results of the selective attention processes for both the FPGA and PC, we used the same image which is extracted from the silicon retina. Even though some distortion is occurred in the processing, the results are almost the same in the data normalization phase. Also, we tested 20 times to measure the processing time. While the computation time for the saliency map process using the PC which has Pentium with dual core CPU and 2G RAM is around 10 msec and 40 msec for release mode and debug mode, respectively, the FPGA's processing time is around 4 msec. According to the result, the FPGA processing time is faster than PC.

We estimated and compared the power consumption of the proposed system and that of the system with a PC. While the power consumption of the silicon retina with FPGA was about 1.5 W, that of CPU alone in the PC was estimated to be about 49 W; this value was estimated using Sandra lite [10]. The power dissipation of the proposed system is much lower than that with a PC.

5   Conclusion

We proposed a hardware oriented vision system having a biologically inspired visual selective attention function. The proposed system was implemented based on a

neuromorphic silicon retina chip for obtaining visual stimuli. And we implemented the selective attention algorithm using the FPGA module. In FPGA, we simplified the bottom-up saliency map model for the compactness. We devised an efficient center-surround processing algorithm for properly implementing the algorithm in a limited hardware system with low capacity memory units. We also reduced normalization distortion by employing an integer denominator. The preprocessing in the silicon retina and the parallel processing in the FPGA reduced the processing time as well as power consumption compared to the system with a PC. The selective attention results of the proposed system was similar to that of the original algorithm which uses a PC. As further work, we will implement higher level functions of the brain like top-down visual selective attention. And we are also considering applying the proposed system to real world applications like surveillance problem.

References

1. Itti, L., Koch, C., Neibur, E.: A model of saliency–based visual attention for rapid scene analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(11), 1254–1259 (1998)
2. Koike, T., Saiki, J.: Stochastic guided search model for search asymmetries in visual search tasks. In: Proceedings of Biologically Motivated Computer Vision, Tübingen, pp. 408–417 (2002)
3. Kadir, T., Brady, M.: Scale, saliency and image description. Int. J. Comput. Vis. 45(2), 83–105 (2001)
4. Ramström, O., Christensen, H.I.: Visual Attention Using Game Theory. In: Proceedings of Biologically Motivated Computer Vision, Tübingen, pp. 462–471 (2002)
5. Jeong, S., Ban, S.-W., Lee, M.: Stereo saliency map considering affective factors and selective motion analysis in a dynamic environment. Neural Netw. 21, 1420–1430 (2008)
6. Ban, S.-W., Jang, Y.-M., Lee, M.: Affective saliency map considering psychological distance. Neurocomputing 74(11), 1916–1925 (2011)
7. Indiveri, G.: Neuromorphic VLSI Models of Selective Attention: From single Chip Vision Sensors to Multi-chip Systems. Sensors 8, 5352–5375 (2008)
8. Goldstein, E.B.: Sensation and perception, 4th edn. An International Thomson Publishing Company, USA (1996)
9. Itti, L., Koch, C.: Feature combination strategies for saliency-based visual attention systems. J. Electronic Imaging 10, 161–169 (2001)
10. SiSoftware Sandra Lite, http://www.sisoftware.net/

Event-Driven Simulation of Arbitrary Spiking Neural Networks on SpiNNaker

Thomas Sharp, Luis A. Plana, Francesco Galluppi, and Steve Furber

School of Computer Science, The University of Manchester,
Oxford Road, Manchester, UK
thomas.sharp@cs.man.ac.uk

Abstract. Programming supercomputers correctly and optimally is non-trivial, which presents a problem for scientists simulating large areas of the brain. Researchers face the challenges of learning how to fully exploit hardware whilst avoiding the numerous pitfalls of parallel programming such as race conditions, deadlock and poor scaling. The SpiNNaker architecture is designed to exploit up to a million processors in modelling as many as one billion neurons in real-time. We present a programming interface for the architecture to allow modelling of arbitrary neuron and synapse dynamics using standard sequential C code, without concern for parallel-programming techniques or interprocessor communication mechanisms. An example is presented in which SpiNNaker is programmed to model multiple synaptic dynamics that are exchanged on the fly and the results of the different synaptic efficacies are shown.

Keywords: neural simulation, event driven, parallel programming, SpiNNaker, kernel, tasks, callbacks, C.

1 Introduction

Parallel computers are powerful tools for modelling large-scale, biologically-plausible Spiking Neural Networks (SNNs) [4]. However, the energy requirements of real-time brain-scale simulations in digital circuits [8] are prohibitively expensive and are expected to remain so for some decades [6]. *Neuromorphic engineering* [11] has produced analogue circuits which simulate neuron and synapse dynamics with great speed and energy efficiency, but at the expense of model adaptation in light of discoveries in 'wet' and computational neurosciences.

SpiNNaker is a digital many-core hardware architecture that aims to address these issues using low-power, general-purpose processors. The SpiNNaker operating system kernel must abstract the details of the hardware to allow researchers to easily simulate arbitrary neuron and synapse models whilst maintaining real-time performance and power efficiency. To this end, inspiration is taken from microcontroller kernels which operate under strict timing and energy requirements [5]. This paper describes event-driven computation as a solution to the problem of large-scale parallel-programming, presents the SpiNNaker Application Run-Time Kernel (ARK) and Application Programming Interface (API) and shows their use in modelling a number of synaptic dynamics.

2   SpiNNaker

SpiNNaker is a massively-parallel computing architecture designed to model billion-neuron, trillion-synapse SNNs in real-time. A SpiNNaker machine consists of up to 2^{16} multiprocessor chips (figure 1) containing eighteen low-power ARM processor cores dedicated to simulation of up to 10^3 neurons and 10^6 synapses each [10]. Processors communicate neural spikes and other simulation data via an on-chip router that also forms links with six neighbouring chips so that any processor may communicate with any other in the machine.

Instruction memory
run-time kernel
application callbacks

Router
routing tables
spike packet routing
system comms.

Data memory
kernel state
neuron states
stack and heap

RAM port
synapse states
activity logs

Processor
neuron and synapse
state computations
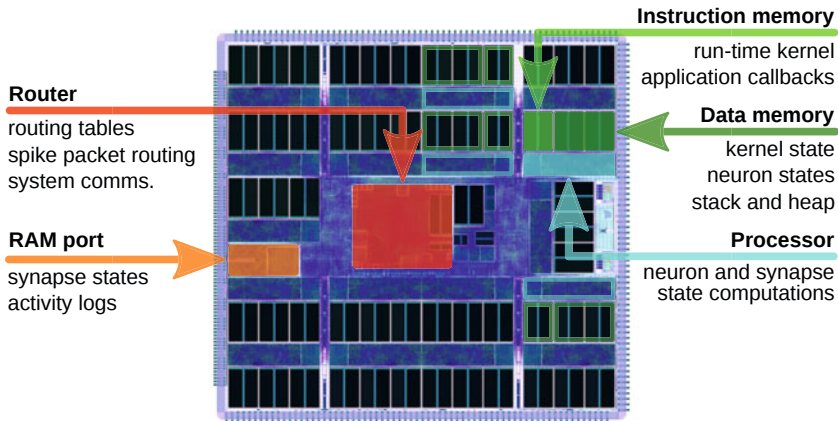
Fig. 1. A SpiNNaker chip. Each processor has interrupt control and timer peripherals and 32kB instruction and 64kB data memories. A shared 128MB off-chip memory is accessed by DMA transfers. A packet-switched router handles communications between processors on local and remote chips.

3   Event-Driven Neural Simulation

SpiNNaker applications are *event-driven* (figure 2) in that all computational *tasks* follow from events in hardware. Neuron states are computed in discrete timesteps initiated in each processor by a local periodic *timer event*. At each timestep processors evaluate the membrane potentials of all of their neurons given prior synaptic inputs and deliver a packet to the router for each neuron that spikes. Spike packets are routed to all processors that model neurons efferent to the spiking neuron. Receipt raises a *packet event* that prompts the efferent processor to retrieve the appropriate synaptic weights from off-chip RAM using a background Direct Memory Access transfer. The processor is then free to perform other computations during the DMA transfer and is notified of its completion by a *DMA done event* that prompts calculation of the sizes of synaptic inputs to subsequent membrane potential evaluations.
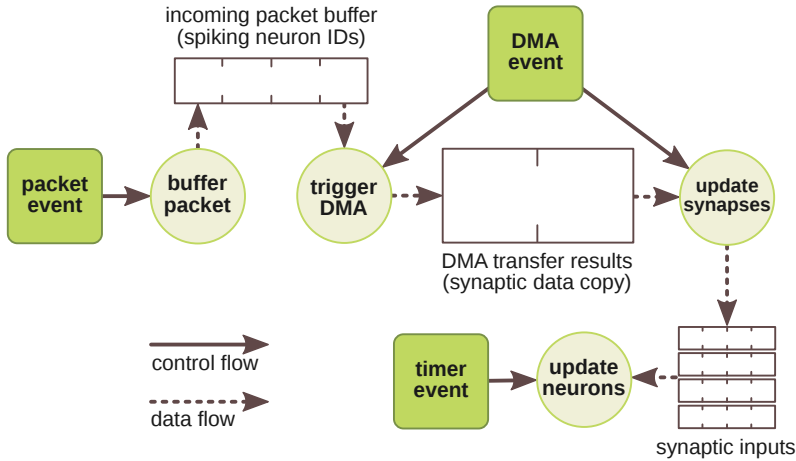
Fig. 2. Events and corresponding tasks in a typical neural simulation

Each SpiNNaker processor executes an instance of the Application Run-Time Kernel (ARK) which is responsible for providing computational resources to the tasks arising from events. The ARK has two threads of execution (figure 3) that share processor time: following events, control of the processor is given to the *scheduler* thread that queues tasks; upon its completion, the scheduler returns control to the *dispatcher* thread that dequeues tasks and executes them. In terms of figure 2, for example, a timer event schedules a neuron update task that is dispatched upon returning from the event.

Tasks have *priorities* that dictate the order in which they are executed by the dispatcher. The scheduler places each task at the end of the queue corresponding to its priority and the dispatcher continually executes tasks from the highest-priority non-empty queue. To facilitate immediate execution, priority zero tasks are *non-queueable* and are executed by the scheduler directly, precluding any further scheduling or dispatching until the task is complete.

The SpiNNaker Application Programming Interface (API) allows a user to specify the tasks that are executed following an event. The user writes *call-back* functions in C that encode the desired tasks and then registers them with the scheduler against particular events. The following example lists callbacks to compute the Izhikevich equations (see [9] and [10] for details) on the timer event, to buffer packets and kickstart DMA transfers on a packet event and to start subsequent DMA transfers (conditional on receipt of further packets) and process synaptic inputs on the DMA done event. Three variants of the timer callback are provided which compute different dynamics of synaptic efficacy, namely, current-based instantaneous spike response synapses and current- and conductance-based synapses with first-order response dynamics [3, 1]. It should be noted that these models serve only to demonstrate the API; readers are referred to the citations for an explanation of the neural activity itself.
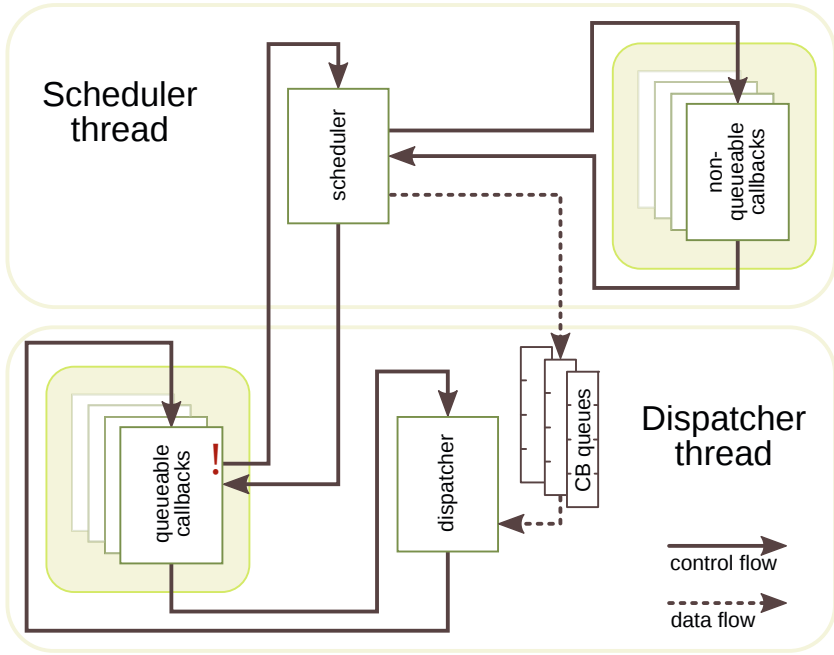
Fig. 3. Control and data flow between the scheduler and dispatcher threads

In the `main` function the timer callback of the first synapse model is registered along with the packet and DMA done callbacks. A simulation of a single neuron receiving input spikes every 16 milliseconds is run for 800 milliseconds. At $t = 400$ms and $t = 600$ms the timer callback registers a new callback in the scheduler, which causes a new set of synaptic dynamics to be computed on subsequent timer events. The code for functions provided by the API is not listed and for brevity only excitatory synaptic inputs (buffers denoted `epsp`) are shown.

```
int main() {
  // Call hardware and simulation configuration functions
  ...
  // Register callbacks and run simulation
  callback_on(PACKET_EVENT, packet_callback, PRIORITY_1);
  callback_on(DMA_DONE_EVENT, dma_done_callback, PRIORITY_2);
  callback_on(TIMER_EVENT, timer_callback_0, PRIORITY_3);
  start(800);
}

void feed_dma_pipeline() {
  // Start engine if idle and transfers pending
  if(!dma_busy() && !dma_queue_empty()) {
    void *source = lookup_synapses(packet_queue_get());
    dma_transfer(..., source, ...);
  }
}

void buffer_post_synaptic_potentials(synapse_row_t *synapse_row) {
  for(uint i = 0; i < synapse_row_length; i++) {
    // Get neuron ID, connection delay and weight for each synapse
```

```
    ...
    // Store synaptic inputs
    neuron[neuron_id].epsp[connection_delay] += synaptic_weight;
  }
}

void dma_done_callback(uint synapse_row, uint unused) {
  // Restart DMA engine if transfers pending
  feed_dma_pipeline();
  // Deliver synaptic inputs to neurons
  buffer_post_synaptic_potentials((synapse_row_t *) synapse_row);
}

void packet_callback(uint key, uint payload) {
  // Queue DMA transfer and start engine if idle
  packet_queue_put(key);
  feed_dma_pipeline();
}

void timer_callback_0(uint time, uint null) {
  if(time >= 400) callback_on(TIMER_EVENT, timer_callback_1, PRIORITY_3);
  for(int i = 0; i < num_neurons; i++) {
    uint current = neuron[i].epsp[time];
    // Compute neuron state given input and deliver spikes. See Jin et al.
    ...
    if(neuron[i].v > THRESHOLD){
      send_mc_packet(neuron[i].id);
    }
  }
}

void timer_callback_1(uint time, uint null) {
  if(time >= 600) callback_on(TIMER_EVENT, timer_callback_2, PRIORITY_3);
  for(int i = 0; i < num_neurons; i++) {
    uint current = neuron[i].epsp[time];
    // Compute neuron state given input and deliver spikes.
    ...
    if(neuron[i].v > THRESHOLD){
      send_mc_packet(neuron[i].id);
    }
    // Add an exponentially decaying quantity to next timestep's input
    neuron[i].epsp[time + 1] += current * neuron[i].decay;
  }
}

void timer_callback_2(uint time, uint null) {
  for(int i = 0; i < num_neurons; i++) {
    // Get synaptically induced conductance (scaled down)
    int conductance = neuron[i].epsp[time] / 64;
    // Compute current from conductance and membrane and eq. potentials
    int current = conductance * (neuron[i].v - EQUILIBRIUM_POTENTIAL);
    // Compute neuron state given input and deliver spikes.
    ...
    if(neuron[i].v > THRESHOLD){
      send_mc_packet(neuron[i].id);
    }
    // Add an exponentially decaying quantity to input at time + 1
    neuron[i].epsp[time + 1] += current * neuron[i].decay;
  }
}
```

The effect of each of the synapse models on the input terms and neuron membrane potential is shown in figure 4. Spikes are clipped from the top of the figure and synaptic currents are negatively offset for clarity. The instantaneous synapse response elicits a brief spike in the membrane potential of the neuron but does not provide enough drive to cause a spike. The first-order dynamics of

the next synapse model cause the neuron to integrate significantly more current and thus to spike. The neuron also spikes under the third synapse model and interactions between the membrane potential and input current are visible, such as where the membrane potential approaches 0 and the current term decreases correspondingly.
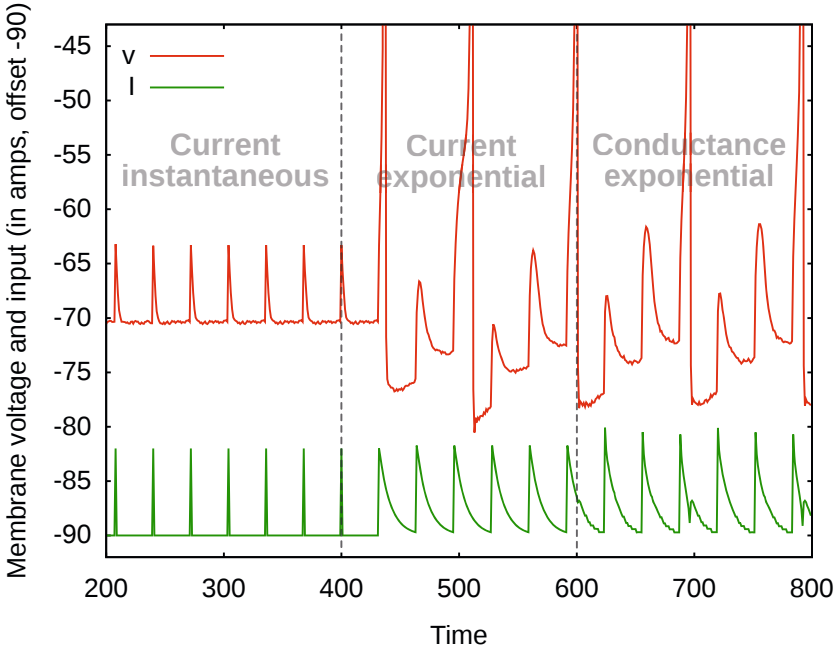
Fig. 4. Membrane potential and input current traces for three synapse models

4 Conclusion

Parallel computers are powerful tools in neuroscientific research but programming them is non-trivial. Researchers face two particular challenges to exploiting supercomputers: the difficulty of parallel programming, in terms of both program optimality and correctness; and knowledge of the numerous capabilities and communication mechanisms of parallel computing hardware. The Application Run-Time Kernel and Application Programming Interface presented in this paper largely addresses these concerns; researchers use template programs and implement just the sequential, standard C code required to compute neuron and synapse dynamics or borrow from a repository of established models. Network structure (the number of neurons and their connectivity) is then specified at a desktop computer using PyNN [2] from which the simulation data structures are compiled and loaded into the machine [7]. Parallelism is achieved transparently by executing sequential programs in numerous processors simultaneously and interprocessor communication is handled in terms of neural spikes, which are

transmitted by an API function call and received by a callback registered against the packet received event in the ARK. Thus, a model is provided for exploiting a massively-parallel computer for real-time simulation of large-scale spiking neural networks without the otherwise major challenges of ensuring correctness and optimality of large parallel programs.

References

1. Brette, R.: Exact simulation of integrate-and-fire models with synaptic conductances. Neural Computation 18, 2004–2027 (2006)
2. Davidson, A.P., Brüderle, D., Eppler, J.M., Kremkow, J., Muller, E., Pecevski, D., Perrinet, L., Yger, P.: PyNN: a common interface for neuronal network simulators. Frontiers in Neuroinformatics 2, 1–10 (2009)
3. Dayan, P., Abbott, L.F.: Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems, 1st edn. The MIT Press (2001)
4. de Garis, H., Shuo, C., Goertzel, B., Ruiting, L.: A world survey of artificial brain projects, part I: Large-scale brain simulations. Neurocomputing 74, 3–29 (2010)
5. Emerson, M., Neema, S., Sztipanovits, J.: Handbook of Real-Time and Embedded Systems. CRC Press (2006)
6. Furber, S., Temple, S.: Neural systems engineering. Journal of the Royal Society Interface 4, 193–206 (2006)
7. Galluppi, F., Rast, A., Davies, S., Furber, S.: A General-Purpose Model Translation System for a Universal Neural Chip. In: Wong, K.W., Mendis, B.S.U., Bouzerdoum, A. (eds.) ICONIP 2010, Part I. LNCS, vol. 6443, pp. 58–65. Springer, Heidelberg (2010)
8. Gara, A., Blumrich, M.A., Chen, D., Chiu, G.L.T., Coteus, P., Giampapa, M.E., Haring, R.A., Heidelberger, P., Hoenicke, D., Kopcsay, G.V., Liebsch, T.A., Ohmacht, M., Steinmacher-Burow, B.D., Takken, T., Vranas, P.: Overview of the Blue Gene/L system architecture. IBM Journal of Research and Development 49, 195–212 (2005)
9. Izhikevich, E.M.: Simple model of spiking neurons. IEEE Transactions on Neural Networks 14, 1569–1572 (2003)
10. Jin, X., Furber, S.B., Woods, J.V.: Efficient modelling of spiking neural networks on a scalable chip multiprocessor. In: International Joint Conference on Neural Networks, pp. 2812–2819 (2008)
11. Mead, C.A.: Analog VLSI and neural systems. Addison-Wesley (1989)

Geometry vs. Appearance for Discriminating between Posed and Spontaneous Emotions

Ligang Zhang, Dian Tjondronegoro, and Vinod Chandran

Queensland University of Technology, 2 George Street, Brisbane, 4000, Australia
ligzhang@gmail.com, {dian,v.chandran}@qut.edu.au

Abstract. Spontaneous facial expressions differ from posed ones in appearance, timing and accompanying head movements. Still images cannot provide timing or head movement information directly. However, indirectly the distances between key points on a face extracted from a still image using active shape models can capture some movement and pose changes. This information is superposed on information about non-rigid facial movement that is also part of the expression. Does geometric information improve the discrimination between spontaneous and posed facial expressions arising from discrete emotions? We investigate the performance of a machine vision system for discrimination between posed and spontaneous versions of six basic emotions that uses SIFT appearance based features and FAP geometric features. Experimental results on the NVIE database demonstrate that fusion of geometric information leads only to marginal improvement over appearance features. Using fusion features, surprise is the easiest emotion (83.4% accuracy) to be distinguished, while disgust is the most difficult (76.1%). Our results find different important facial regions between discriminating posed versus spontaneous version of one emotion and classifying the same emotion versus other emotions. The distribution of the selected SIFT features shows that mouth is more important for sadness, while nose is more important for surprise, however, both the nose and mouth are important for disgust, fear, and happiness. Eyebrows, eyes, nose and mouth are important for anger.

Keywords: Facial expression, posed, spontaneous, SIFT, FAP.

1 Introduction

A machine vision system that can accurately discriminate spontaneous from posed expressions can be useful in ways similar to a polygraph. Spontaneous expressions are difficult to distinguish from posed ones and differ in subtle ways in appearance, timing and accompanying head movement [1]. In general, facial expression recognition (FER) can be reliably performed from still images with far less complexity. However, the timing and head movement can only be extracted from video, not from still images. The distances between key points on a face extracted from a still image can indirectly capture some movement and pose changes using active shape models (ASM). This information is superposed on information about non-rigid facial movement that is also part of the expression.

Relatively little research has been conducted on machine discrimination between posed and spontaneous facial expressions. These efforts mainly focus on smile [1], [2], [3], eyebrow action [4] and pain [5], [6]. For smile and eyebrow action, nearly all known approaches are designed based on the movements of facial points. Hamdi *et al.* [2] used eyelid movements and reported 85% and 91% accuracy in discriminating between posed and spontaneous smiles on the BBC and Cohn-Kanade databases, respectively. Michel *et al.* [3] proposed a multimodal system to discern posed from spontaneous smiles by fusing a set of temporal attributes of tracked points of face, head and body. 94% accuracy was the best result, obtained with late fusion of all modalities. Michel *et al.* [4] also proposed to use the temporal dynamics of facial points to distinguish between posed and spontaneous brow actions, and attained a 90.7% classification rate. Littlewort *et al.* [5] employed a two-stage system to differentiate faked pain from real pain: a detection stage for 20 facial actions using Gabor features and a SVM classification stage, achieving 88% accuracy. Bartlett *et al.* [6] reported 72% classification accuracy of posed versus spontaneous pain through Gabor feature based facial action detection. Other facial expressions, such as anger, disgust, fear, sadness, and surprise have not been fully investigated in this context.

Appearance features (e.g. SIFT and Gabor) are more suitable for capturing the subtle changes of the face; while geometric features (e.g. distance between landmarks) are more capable of representing shape and location information of facial components. Although the fusion of appearance and geometry leads to significant performance improvements on basic facial expression classification [7], it remains unclear whether such improvements are possible for posed versus spontaneous emotion discrimination as well. This paper addresses these areas.

An automatic system to distinguish posed from spontaneous versions of six basic emotions using appearance (SIFT) and geometric (FAP) features is adopted to investigate recognition performance. Feature selection is performed using minimal redundancy maximal relevance (mRMR) and classification using a support vector machine (SVM). Scale-invariant feature transform (SIFT) has been shown a better recognition performance of facial expressions in previous work [8] than other appearance features, including LBP and HOG, while the distances defined based on facial animation parameters (FAPs) have also been demonstrated as a sparse, compact, yet information-rich representation of the facial shape [9]. But they have not been combined yet for discriminating posed versus spontaneous emotions. Appearance features are extracted locally around key points while distances between key points are used for the geometric feature set. They are therefore expected to not contain overlapping information. Intuitively, geometry is not expected to result in significant improvement in performance in this context because no temporal information is captured. This paper will compare the relative importance of the two types of features and those extracted from different regions on the face for each of six emotions.

The rest of the paper is organized as follows. Section 2 presents the evaluation system. Section 3 gives the experimental results. Conclusions are drawn in Section 4.

2   Evaluation Framework

Fig. 1 shows the framework of the evaluation system. From an input image, the face region is detected using the widely used Viola-Jones detector, and 68 facial fiducial

points are detected using a well-trained active shape model (ASM). SIFT descriptors are extracted around each of 53 interior facial points. Feature vectors from all points are concatenated into a single vector representing *appearance* features. A subset of the most discriminative appearance features is selected using the mRMR algorithm. *Geometric* features composed of 43 distances defined using an active shape model (ASM) and FAPs are also extracted. The normalized *appearance* feature subset and *geometric* features can be used alone or combined through a feature-level fusion. A SVM with a radial basis function (RBF) kernel is used as the classifier for discriminating between posed and spontaneous versions of six basic emotions - anger (AN), disgust (DI), fear (FE), happiness (HA), sadness (SA) and surprise (SU).
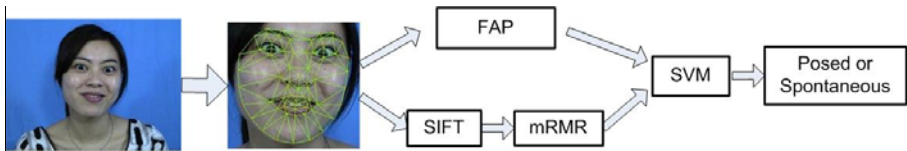
Fig. 1. Framework of the evaluation system

2.1 Face and Fiducial Point Detection

Once the face region is detected by the Viola-Jones detector, an ASM [10] is used to detect the fiducial points. To train the ASM, we collected 100 images from the internet with different natural emotions and different face poses ranging from -20 to 20 degrees. Then 68 fiducial points as shown in Fig.2a are manually annotated with x and y locations. The trained ASM is expected to work well on faces with normal face movements. It has been observed that the points on the face boundary (Index from 1 to 15 in Fig.2a) are not always accurately detected due to facial shape changes between subjects and facial movements (as shown in Fig.2b). Further, the regions around these points contain background information and do not provide reliable features. Therefore, only 53 interior points (Index from 16 to 68 in Fig.2a) are used to extract SIFT features.

(a)   (b)
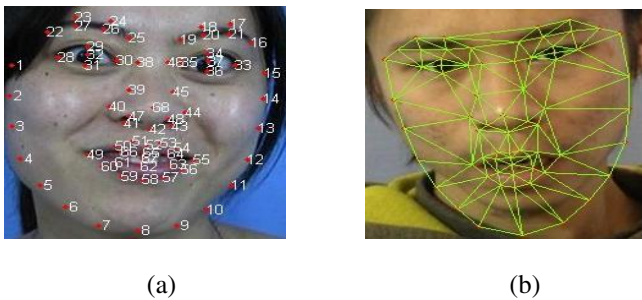
Fig. 2. (a) 68 fiducial points for training ASM and (b) detection results with inaccurate boundary points

2.2   SIFT Feature Extraction

SIFT [11] provides distinctive invariant features suitable for detecting salient key points and describing local appearance. The SIFT is known to be invariant to image scale and rotation, and robust across a substantial range of affine distortion, changes in 3D viewpoint, noise and illumination. SIFT features extracted around a small set of facial landmarks have been applied to describe local characteristics of the face, yielding promising results [12]. SIFT features around a number of points also helps to achieve a degree of tolerance to face movements and pose changes.

Following the settings in [12], the SIFT descriptor is computed from the gradient vector histograms of the pixels in a 4×4 patch around each point of 53 interior points. Instead of setting a fixed orientation, we let the program compute the 8 possible gradient orientations. Therefore, each SIFT descriptor contains a total of 128 elements. By computing one such descriptor at each point, we obtain a final feature vector with 6,784 elements.

2.3   Geometric Feature Extraction

Facial animation parameters (FAPs) [13] are defined in the ISO MPEG-4 standard (part 2, visual) to allow the animation of synthetic face models. FAPs contain 68 parameters that are either high level parameters describing visemes and expressions, or low level parameters describing displacements of the single points of the face. Therefore, FAPs can provide a concise representation of the evolution of the expression of the face and can represent a complete set of basic facial actions. Furthermore, FAPs also can handle arbitrary faces through the use of FAP units (FAPUs), which are defined as the fractions of distances between key points.

Geometric features include 43 distances between the 53 interior points. As listed in Table 1, these distances are calculated based on FAPs to allow the animation of face shape changes. Compared with facial movement vectors in multi-frames, distance features have the merit of being robust to pose changes, and do not require

Table 1. Distances between facial points defined by FAPs

No.	Distance	No.	Distance	No.	Distance	NO.	Distance
3	Dy(52,58)	19	Dy(29,32)	33*	Dy(32,27)	55	Dy(50,42)
4	Dy(65,42)	20	Dy(34,37)	34*	Dy(37,17)	56	Dy(54,42)
5	Dy(62,42)	21	Dy(31,32)	34*	Dy(37,18)	57	Dy(60,42)
6	Dx(49,42)	22	Dy(36,37)	34*	Dy(37,20)	58	Dy(56,42)
7	Dx(55,42)	29	Dy(29,31)	34*	Dy(37,21)	61*	Dx(30,40)
8	Dy(66,42)	30	Dy(34,36)	35	Dy(28,22)	61*	Dx(30,39)
9	Dy(64,42)	31	Dy(30,25)	36	Dy(33,16)	62*	Dx(35,44)
10	Dy(61,42)	32	Dy(35,19)	37	Dx(30,25)	62*	Dx(35,45)
11	Dy(63,42)	33*	Dy(32,23)	38	Dx(35,19)	63	Dy(35,68)
12	Dy(49,42)	33*	Dy(32,24)	51	Dy(52,42)	64	Dx(35,68)
13	Dy(55,42)	33*	Dy(32,26)	52	Dy(58,42)	-	-

Note: Dx(M,N) and Dy(M,N) indicate the distances between two points indexed M and N in the horizontal and vertical directions respectively. M and N are based on the 53 interior points in Fig. 2a.

compensation for face movements. The distances defined based on FAPs have been demonstrated as a sparse, compact, yet information-rich representation of the facial shape [9]. Therefore, they are suitable for the proposed system. To allow for variations between faces, FAPUs are defined as the fractions of distances between key points to scale FAPs (i.e. 43 distances).

2.4 Discriminative Texture Feature Selection

We use the minimal redundancy maximal relevance criterion (mRMR) [14] algorithm to select a subset of the most discriminative features from the extracted SIFT features. The mRMR selects a subset of features that jointly have the largest dependency on the ground truth class and the least redundancy among the features, according to following equation:

$$\max_{x_j \in X - S_{m-1}} [I(x_j; c) - \frac{1}{m-1} \sum_{x_i \in S_{m-1}} I(x_j, x_i)] \tag{1}$$

Where x_i and x_j are the ith and j^{th} features of the feature set X; I(x,y) is the mutual information of two variables x and y; c are the class labels. Suppose S_{m-1} is the already selected feature set with m-1 features, the task is to select the mth feature from the set $\{X- S_{m-1}\}$.

When using mRMR, a discretization of the continuous inputs was recommended [14]. This paper obtains the discrete feature $\overline{D}_k$ of a continuous feature D_k based on the mean value μ_k and the standard deviation σ_k of all features:

$$\overline{D}_k = \begin{cases} -2 & if & D_k < \mu_k - \sigma \cdot \sigma_k \\ 0 & if & \mu_k - \sigma \cdot \sigma_k \le D_k \le \mu_k + \sigma \cdot \sigma_k \\ 2 & if & D_k > \mu_k + \sigma \cdot \sigma_k \end{cases} \tag{2}$$

Where σ is set to 0.5.

3 Experiments

3.1 Database

The natural visible and infrared facial expression (NVIE) database [16] is a newly developed comprehensive platform for both spontaneous and posed facial expression analysis. The spontaneous expressions are induced by film clips deliberately selected from the internet, while the posed ones are obtained by asking the subjects to perform a series of expressions. There are a total of 215 healthy students (157 males and 58 females), ranging in age from 17 to 31. Among them, 105, 111, 112 subjects participated in the spontaneous database under front, left and right illumination respectively, and 108 subjects participated in the posed database. Both spontaneous and posed images with peak emotions are labeled with six basic emotions.

(a)   (b)
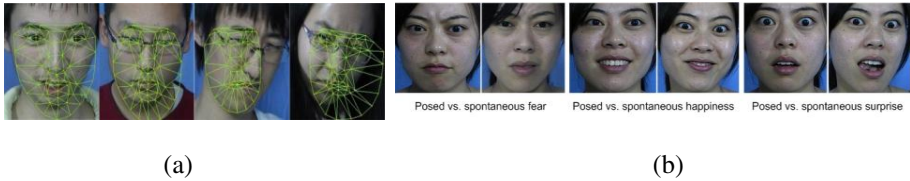
Fig. 3. (a) Image samples excluded from the experiment because of inaccurate detection results by ASM. (b) Images samples of posed versus spontaneous emotions.

Table 2. Distribution of the used images over six emotions

	AN	DI	FE	HA	SA	SU
Posed	593	578	609	607	581	604
Spontaneous	229	266	211	315	236	215

In this paper, all posed peak visible images are used, while only spontaneous visible images with final evaluated annotations are used. Note that a part of spontaneous images have not been provided with final annotations by the time of writing this paper. After removing those failed during face and facial point detection, we get 3,572 posed and 1,472 spontaneous images. Fig. 3a shows samples of removed images due to inaccurate ASM detection results of facial points. As can be seen, the mouth region and the face boundary are less likely to be accurately detected by the ASM when big out-of-plane rotations of the face occur (i.e. pitch and yaw). Fig. 3b demonstrates samples of posed versus spontaneous emotions and Table 2 shows the distribution of the images over six emotions.

3.2   Classification Performance

We conducted subject-independent tests to obtain an average classification result over 10 cross-validations. In each cross-validation, images of 10% subjects are randomly selected for testing and the images of 90% subjects left are for training. The process repeats 10 times to obtain average classification accuracy. Note that the emotional labels of each of the six basic emotions are assumed to be known before classifying posed versus spontaneous emotions using a SVM.

Fig. 4 shows the accuracy of posed versus spontaneous classification of six emotions. As anticipated, fusion of SIFT and FAP features only leads to a marginally higher overall performance than using SIFT features only, for all emotions except for disgust. For disgust, inclusion of FAP features in fact leads to a lower performance than using SIFT features alone. The use of FAP features does not improve the performance, and this may be because FAP based distances have a limited capacity to capture the temporal information (e.g. movements) of facial expressions, while discrimination between posed and spontaneous emotions largely depends on such information as shown in previous studies [1], [17], [18]. The results agree with the claim in [18] that high-abstraction features extracted from video segments can capture more general physical phenomena than low-abstraction features in one frame. In addition, posed and spontaneous emotions in static images are more likely to have
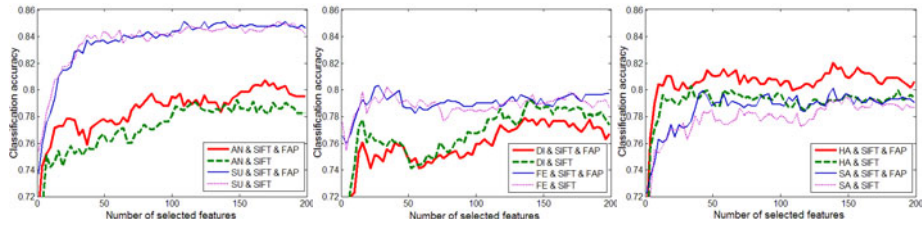
Fig. 4. Classification accuracy for posed versus spontaneous versions of six basic emotions on the NVIE database. As can be seen, fusion of SIFT and FAP only leads to marginally higher overall performance than using SIFT features only. The accuracy of using FAP features alone can be observed from the third row in Table 3. For disgust, fusion is in fact catastrophic.

Table 3. Posed/Spontaneous classification accuracy (%) + one standard deviation

	AN	DI	FE	HA	SA	SU
SIFT+FAP	77.2±6.3	76.1±6.1	79.7±6.2	80.5±6.1	79.7±5.1	83.4±5.2
SIFT	75.6±7.1	76.1±6.9	79.6±7.3	79.3±7.7	77.4±7.6	83.9±4.0
FAP	71.2 ±4.9	69.7±6.0	76.2±4.0	65.6±5.8	68.7±4.8	73.3±6.3

similar geometric distances, and their differences are mainly conveyed by subtle appearance features. It should be noted that a subset of facial points located by ASM without enough precision could also introduce noise to FAP features.

The adopted system using SIFT+FAP or SIFT features obtains an accuracy of more than 74% for all emotions. Among the six emotions, surprise and happiness are the two easiest ones to be distinguished as posed or spontaneous, whereas disgust is the most difficult one. The results are similar to those obtained on recognition of six basic emotions, in which surprise and happiness are often the easiest emotions to recognize, while disgust is one of the most difficult ones. It also can be observed from Fig. 4 that anger, fear and sadness have a similar performance.

Table 3 demonstrates the classification accuracy for posed versus spontaneous emotions based on 40 SIFT and 43 FAP features. Using SIFT+FAP features, the employed system obtains the highest accuracy of 83.4% when testing on surprise, and the lowest accuracy of 76.1% when testing on disgust. SIFT+FAP and SIFT have a similar performance for most of the six emotions, and they both outperform using FAP features alone. For anger, disgust, fear, and surprise, the performances of SIFT+FAP, SIFT and FAP are not statistically significantly different as observed when one standard deviation intervals are noted in Table 3. For happiness and sadness, the performance of SIFT+FAP is also similar to SIFT, but is one standard deviation more significant than FAP. This result again implies that geometric features play a less important role than appearance features on posed versus spontaneous emotion discrimination, and including geometric features leads to little performance improvements compared with using appearance features alone.

3.3 Feature Importance Comparison

To investigate the importance of different points in their contribution to discrimination, we display the distribution of the selected SIFT features over the 53

<p style="text-align:center">(a)   (b)</p>
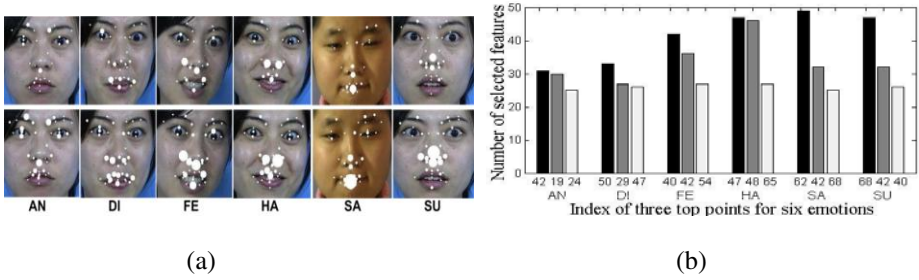
Fig. 5. (a) Distribution of selected SIFT features over 53 points. The two rows contain the top 40 and top 90 features selected by the mRMR in each of 10 subject-independent tests. Emotion class labels are given below. (b) The three top facial points selected for each of the six emotions. Index numbers are mapped to face locations in Fig. 2a.

interior points as shown in Fig.5a. If more features are selected from a given facial point, that point may be considered more important and is marked by a bigger white circle. It should be noted that similar distributions have also been observed using different subsets of the NVIE data.

We can observe that different facial elements play different roles in distinguishing different posed versus spontaneous emotions. The mouth appears to be more important for sadness, the nose region is more important for surprise, while both the nose and mouth are important for disgust, fear, and happiness. The eyebrows, eyes, nose and mouth all play a significant role for anger. The results are contrary to the findings in discriminating six emotions in thermal images [16], where the mouth region has the smallest impacts on all six emotions, and the nose has little impacts on sadness, surprise, fear, and happiness. However, one common point between our work and [16] is that the nose plays a significant role in classifying anger versus other emotions, and discriminating posed versus spontaneous anger. Similar importance of nose is also found for disgust. The important role of the nose in the evaluated framework contrasts with the common understanding that the nose is the relative expression-invariant facial region. Therefore, there appears to exist different important facial regions between discriminating posed versus spontaneous version of one emotion and classifying the same emotion versus other emotions. This is within our expectation as each emotion has its own discriminative facial regions when classifying it versus other emotions. However, discriminating posed versus spontaneous versions of the same emotion needs to depend on information in other regions.

From Fig. 5a, we also can see that feature points on the eyebrows and eyes seem to provide few of the top 40 or top 90 features, for most of the emotions. This is probably due to the fact that about a half of the faces in the NVIE database have glasses, which occlude the useful information in the eyebrows and eyes. In addition, feature points on the mouth also have different distributions for different emotions. For instance, the points focus on the corner lip for fear, the top lip for happiness, and the middle lip for sadness.

Fig.5b gives the three top facial fiducial points that contain the largest number of the selected features for each emotion. As can be seen, most of these points for six emotions are distributed on nose and the points indexed 42, 40, 47, 68 are shared by different emotions (e.g. the point 40 is shared by fear and surprise). Compared with

the top points for other emotions, the point 62 for sadness and the point 68 for surprise take a much larger proportion of the selected features.

4   Conclusions

A machine vision system for distinguishing posed and spontaneous versions of six basic emotions in static images is used to compare the performance of SIFT features from ASM based fiducial points, FAP distance features and their fusion. Experimental results show that appearance features play a significantly more important role than geometric features on posed versus spontaneous emotion discrimination, and fusion of geometric information leads only to marginal improvement over SIFT appearance features. This is owing to the fact that temporal information is not available in the geometric representation of still images. Among six emotions, surprise is the easiest emotion (83.4% accuracy) to be classified as posed or spontaneous, while disgust is the most difficult one (76.1%) using SIFT+FAP features. Our results find that there are different important facial regions between discriminating posed versus spontaneous version of one emotion and classifying the same emotion versus other emotions. In terms of providing the most relevant features for classification between posed and spontaneous emotions, the mouth is more important for sadness, the nose is more important for surprise, while both the nose and mouth are important for disgust, fear, happiness, and the eyebrows, eyes, nose, mouth are all important for anger. A significant proportion of the SIFT features selected by the mRMR for six emotions are distributed on the points in the nose region. Our future work will test the performance fusing SIFT features with temporal geometric features in video, and explore real world applications.

References

1. Cohn, J., Schmidt, K.: The Timing of Facial Motion in Posed and Spontaneous Smiles. International Journal of Wavelets, Multiresolution and Information Processing 2, 1–12 (2004)
2. Hamdi, D., Roberto, V., Albert Ali, S., Theo, G.: Eyes Do Not Lie: Spontaneous versus Posed Smiles. In: Proceedings of the International Conference on Multimedia, pp. 703–706. ACM, Firenze (2010)
3. Michel, F.V., Hatice, G., Maja, P.: How to Distinguish Posed from Spontaneous Smiles Using Geometric Features. In: Proceedings of the 9th International Conference on Multimodal Interfaces, pp. 38–45. ACM, Nagoya (2007)
4. Michel, F.V., Maja, P., Zara, A., Jeffrey, F.C.: Spontaneous vs. Posed Facial Behavior: Automatic Analysis of Brow Actions. In: Proceedings of the 8th International Conference on Multimodal Interfaces, pp. 162–170. ACM, Banff (2006)
5. Littlewort, G.C., Bartlett, M.S., Lee, K.: Automatic Coding of Facial Expressions Displayed During Posed and Genuine Pain. Image and Vision Computing 27, 1797–1803 (2009)

6. Bartlett, M., Littlewort, G., Vural, E., Lee, K., Cetin, M., Ercil, A., Movellan, J.: Data Mining Spontaneous Facial Behavior with Automatic Expression Coding. In: Esposito, A., Bourbakis, N.G., Avouris, N., Hatzilygeroudis, I. (eds.) HH and HM Interaction. LNCS (LNAI), vol. 5042, pp. 1–20. Springer, Heidelberg (2008)
7. Mingli, S., Dacheng, T., Zicheng, L., Xuelong, L., Mengchu, Z.: Image Ratio Features for Facial Expression Recognition Application. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 40, 779–788 (2010)
8. Yuxiao, H., Zhihong, Z., Lijun, Y., Xiaozhou, W., Xi, Z., Huang, T.S.: Multi-view Facial Expression Recognition. In: 8th IEEE International Conference on Automatic Face & Gesture Recognition, FG 2008, pp. 1–6 (2008)
9. Hao, T., Huang, T.S.: 3D Facial Expression Recognition Based on Automatically Selected Features. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPRW 2008, pp. 1–8 (2008)
10. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active Shape Models-Their Training and Application. Comput. Vis. Image Underst. 61, 38–59 (1995)
11. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60, 91–110 (2004)
12. Berretti, S., Bimbo, A.D., Pala, P., Amor, B.B., Daoudi, M.: A Set of Selected SIFT Features for 3D Facial Expression Recognition. In: 20th International Conference on Pattern Recognition, ICPR 2010, pp. 4125–4128 (2010)
13. Pandzic, I.S., Forchheimer, R.: MPEG-4 facial animation: the standard, implementation and applications. Wiley (2002)
14. Hanchuan, P., Fuhui, L., Ding, C.: Feature Selection Based on Mutual Information Criteria of Max-dependency, Max-relevance, and Min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence 27, 1226–1238 (2005)
15. Lajevardi, S., Hussain, Z.: Automatic Facial Expression Recognition: Feature Extraction and Selection. Signal, Image and Video Processing, 1–11 (2010)
16. Shangfei, W., Zhilei, L., Siliang, L., Yanpeng, L., Guobing, W., Peng, P., Fei, C., Xufa, W.: A Natural Visible and Infrared Facial Expression Database for Expression Recognition and Emotion Inference. IEEE Transactions on Multimedia 12, 682–691 (2010)
17. Schmidt, K., Ambadar, Z., Cohn, J., Reed, L.: Movement Differences between Deliberate and Spontaneous Facial Expressions: Zygomaticus Major Action in Smiling. Journal of Nonverbal Behavior 30, 37–52 (2006)
18. Schmidt, K., Bhattacharya, S., Denlinger, R.: Comparison of Deliberate and Spontaneous Facial Movement in Smiles and Eyebrow Raises. Journal of Nonverbal Behavior 33, 35–45 (2009)

Towards Learning Inverse Kinematics with a Neural Network Based Tracking Controller

Tim Waegeman and Benjamin Schrauwen

Department of Electronics and Information Systems
Ghent University, Ghent Belgium
http://reslab.elis.ugent.be

Abstract. Learning an inverse kinematic model of a robot is a well studied subject. However, achieving this without information about the geometric characteristics of the robot is less investigated. In this work, a novel control approach is presented based on a recurrent neural network. Without any prior knowledge about the robot, this control strategy learns to control the iCub's robot arm online by solving the inverse kinematic problem in its control region. Because of its exploration strategy the robot starts to learn by generating and observing random motor behavior. The modulation and generalization capabilities of this approach are investigated as well.

Keywords: Adaptive control, Feedback control, Inverse kinematics, Neural network (NN), Reservoir computing (RC).

1 Introduction

Drawing a figure on a blackboard is a task which humans perform without consciously thinking about how each joint of their arm should be positioned. For robots like the iCub [1], this task is much more difficult. A robot needs to be able to map a position from task-space to joint-space, which is called inverse kinematics. There are multiple positions in joint-space to reach a given task-space target. On the other hand, mapping positions from joint-space to task-space, called forward kinematics, is a unique transformation. Solving the inverse kinematics problem has been investigated extensively. Some approaches use analytical and numerical methods to solve this problem [2,3]. More advanced techniques learn inverse kinematics in different sub-regions of the task-space and use a weighting approach to approximate the inverse kinematic over the entire task-space [4,5]. Other techniques use a Recurrent Neural Network (RNN), to train on forward kinematic data containing the positions in joint and task-space [6,7,8]. This removes the redundancy in the inverse transformation, because there are no redundant examples given during training. However, these techniques learn an attractor that is limited to the regions described by the training data and although they posses generalization capabilities, learning the entire inverse kinematic model can not be claimed. The proposed controller learns to control the

iCub's arm online by creating a model based on previous actions and observations of the robot. It uses a Reservoir Computing (RC) network to acquire an inverse kinematic model, but only in the visited regions of the arm. By doing both the control an the updating of the model simultaneously, the desired control behavior can be acquired without the need of having any prior knowledge about the robot.

The remainder of this paper is structured as follows: first, in Section 2, we give a short introduction on Reservoir Computing and explain the training algorithm. Next, in Section 3, the design steps of the controller are explained. To demonstrate the controllers performance, we apply it on a kinematic control task in Section 4. Finally, we draw our conclusions in Section 5.

2   Reservoir Computing

The RC network model used in this paper follows the Echo State Network (ESN) approach [9]. An ESN is composed of a discrete-time recurrent neural network (i.e., the reservoir) and a linear readout output layer which maps the reservoir states to the desired output. For many applications, the dynamics of the reservoir need to be tuned to match the intrinsic time scale of the input data. The system's dynamics can effectively be tuned by using leaky integrator neurons [9]. Their states and the readout output are updated as follows:

$$\mathbf{x}[k+1] = (1-\gamma)\mathbf{x}[k] + \gamma \tanh\left(\mathbf{W}_\mathrm{r}^\mathrm{r}\mathbf{x}[k] + \mathbf{W}_\mathrm{i}^\mathrm{r}\mathbf{u}[k] + \mathbf{W}_\mathrm{b}^\mathrm{r}\right) \tag{1}$$

$$\mathbf{y}[k+1] = \mathbf{W}_\mathrm{r}^\mathrm{o}\mathbf{x}[k+1] + \mathbf{W}_\mathrm{b}^\mathrm{o}, \tag{2}$$

where $\mathbf{u}[k]$ denotes the input at time k, $\mathbf{x}[k]$ represents the reservoir state and $\mathbf{y}[k]$ is the output. The weight matrices $\mathbf{W}_*^\Delta$ represent the connections from $*$ to Δ between the nodes of the network (where r, i, o, b denote *reservoir*, *input*, *output*, and *bias*, respectively). All weight matrices $\mathbf{W}_*^\mathrm{r}$ to the reservoir are initialized randomly, while all connections to the output $\mathbf{W}_*^\mathrm{o}$ are trained using standard linear regression techniques. As non-linearity a hyperbolic tangent function is used. After initialization, $\mathbf{W}_\mathrm{r}^\mathrm{r}$ is normalized by dividing it with its largest absolute eigenvalue (spectral radius). For linear neurons the spectral radius should typically be close, but smaller than one. Because of this spectral radius the system is operating at the edge of stability [9]. The leak rate γ in (1) controls the time scale of the network [9,10]. Usually training $\mathbf{W}_*^\mathrm{o}$ is done offline, in batch mode. In this work, $\mathbf{W}_*^\mathrm{o}$ is trained online using Recursive Least Squares (RLS). With each iteration the output weights are adjusted so that the network converges to the desired output:

$$\mathbf{P}[k] = \mathbf{P}[k-1] - \frac{\mathbf{P}[k-1]\mathbf{x}[k]\mathbf{x}^T[k]\mathbf{P}[k-1]}{(1 + \mathbf{x}^T[k]\mathbf{P}[k-1]\mathbf{x}[k])}, \tag{3}$$

with $\mathbf{P}[0] = \frac{\mathbf{I}}{\alpha}$, $\mathbf{x}[k]$ the current states and α constant. $\mathbf{P}[k]$ is a running estimate of the Moore-Penrose pseudo inverse $(\mathbf{x}^T\mathbf{x} + \rho\mathbf{I})^{-1}$. The used training error is
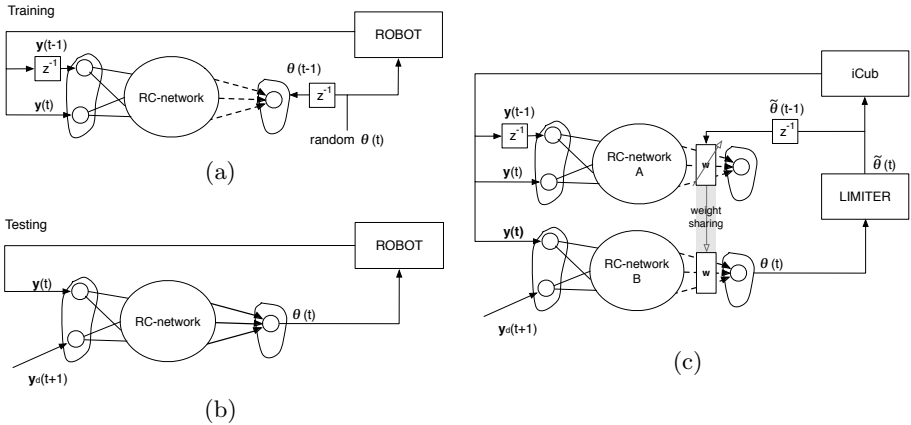
Fig. 1. (a) Illustration of a controller method described in the work of Jaeger [12]. During training, random $\boldsymbol{\theta}(t)$ values are used to train the output weights of the network based on the corresponding robot response $\mathbf{y}(t)$. (b) During testing, the trained network is used to control the robot according to to the desired trajectory $\mathbf{y}_d(t+1)$. (c) Schematic representation of the proposed controller. The dashed arrows represent the output weights $\mathbf{w}$ which are trained. These are the same for both networks (weight sharing). The optional limiter bounds $\boldsymbol{\theta}(t)$ to a certain range. Afterwards, the bounded values $\check{\boldsymbol{\theta}}(t)$ drive the robot. The values $\check{\boldsymbol{\theta}}(t-1)$ are used as desired network output for RC-network A which are used to train the weights $\mathbf{w}$.

defined as the difference between the generated and desired output $d[k]$:

$$e = \mathbf{w}[k-1]\mathbf{x}[k] - d[k] \tag{4}$$
$$\mathbf{w}[k] = \mathbf{w}[k-1] - e\mathbf{P}[k]\mathbf{x}[k]. \tag{5}$$

When using RLS these output weights are rapidly and effectively modified. This behavior satisfies the conditions necessary for the FORCE approach by Sussillo and Abbott [11]. This approach allows learning with feedback of the actual output (small errors included) instead of clamping the feedback to the correct output (no errors) during training.

3 Design of the Controller

The goal of this work is to design a controller which can learn the inverse kinematics in the vicinity of the desired trajectory and without any prior knowledge about the robot system. For most supervised learning techniques, training examples are generated by a teacher controller, observations in joint and task-space or an actual inverse kinematic model of the robot. However, in this work no prior knowledge or model is assumed.

Another approach to learn a good kinematic representation is to use a model exploration strategy where random motor commands are generated and the corresponding robot response is observed. In the work by Jaeger [12], such a strategy

is taken. Here, an RC-network is used which is trained offline by using random values as training output and the plant response to these values as training input. In this example, the feedback information $\mathbf{y}$ is presented to the RC-network in 2 versions: the current feedback $\mathbf{y}(t)$ and a delayed version $\mathbf{y}(t-1)$. During training, also the desired output, which are the random motor commands $\boldsymbol{\theta}(t)$, are delayed one time step before given to the RC-network. As a result, the network learns to generate the previous output $\boldsymbol{\theta}(t-1)$, given the previous $(\mathbf{y}(t-1))$ and current input $(\mathbf{y}(t))$. After training the output weights (dashed lines in Fig. 1(a)), the desired end-effector position $\mathbf{y}_d(t+1)$ is presented to the input which was connected to $\mathbf{y}(t)$ during training. The actual end-effector position on the other hand, is given to the reservoir input which was connected with $\mathbf{y}(t-1)$ during training. The output of the network $\boldsymbol{\theta}(t)$ commands the robot.

In this work we want to achieve similar results but in an online manner. The advantage here is the ability of the controller to readjust the internal model of the robot to unforeseen changes in the robot or its environment during control. As shown in Fig. 1(c), a similar RC-network to the one described above (Fig. 1(a)) is used. This network, which we will call RC-network A, is trained online in a supervised manner by using RLS. Below RC-network A we have a duplicate network, RC-network B, with the same input, reservoir and output weights (weight-sharing) as RC-network A. This network is connected to the robot in a similar manner as described by Jaeger in Fig. 1(b). The output of this network is not only connected to the robot but is also used (delayed by one time step) as the desired output for training the output weights. The reservoir states are initially the same for both networks and are randomly chosen according to a normal distribution ($\mathcal{N}(0,1)$). This random initialization causes the robot to move its arm and generating examples which are spread more evenly over the solution space. Without this so called motor babbling the samples, used to model the kinematics, would be clustered which leads to poor generalization. Because the inputs are not the same for both networks, the corresponding states will evolve differently. However, as RC-network A is converging to a more accurate model, the inputs of both networks will become the same with a difference of 1 time step. Because of the desired trajectory in task-space and the current robot feedback as input, RC-network B starts generating values which are in turn used to command the robot. Such commands are limited to a certain range according to the actuators specifications. For instance, when controlling an actuator the amount of torque that it can deliver is limited. In Fig. 1(c) such limiting is represented by a limiter which bounds $\boldsymbol{\theta}(t)$ to $\tilde{\boldsymbol{\theta}}(t)$. Delayed by one time step, these values $\tilde{\boldsymbol{\theta}}(t-1)$ are given to RC-network A as the desired output. With each iteration, the resulting output weights are used for RC-network B.

By applying this topology, RC-network A is learning the controller solely on the generated examples during actual control. On the contrary, RC-network B uses the trained parameters to improve the control of the robot based on both the desired and actual robot response.
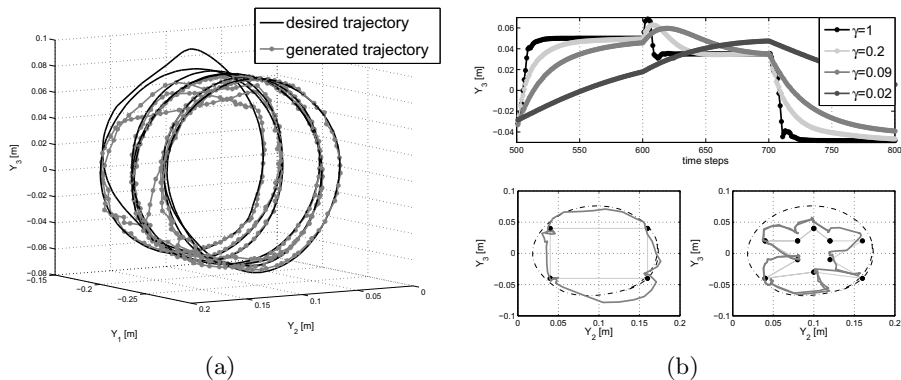
Fig. 2. (a) This plot illustrates a generated trajectory after training (gray) together with the desired trajectory (black). The dots on the generated trajectory are the sample points. (b, top) Demonstrates the velocity modulation on target point reaching (100 time steps per target point) . (b, bottom) Generalization (dark gray) to different target points (black dots). The circular dashed line represents the projection of the spiral on the plain of the target points.

4   Robot Experiments

To evaluate the proposed control strategy we learn a kinematic model of the 7 degree-of-freedom (DOF) iCub robot arm. A "Webots" simulation model of the iCub robot is used to do the experiments. Here Y_1, Y_2 and Y_3 correspond to respectively the Z-, X- and Y-axes of the robot's frame of reference. Encoders in each joint measure the angular positions. When a joint position is given to the robot, an internal PID controller will generate the torque necessary to move the joint to the desired position. Therefore, due to the dynamics of the robot, a delay between the commanded and recorded position is observed.

4.1   RC-Network Setup

RC-network A and B are, except for their input, identical. For choosing the number of neurons, a trade-off between execution speed and performance has to be made. In our experiments we used 400 neurons. All the following parameters are hand tuned. The connection matrix from input to the reservoir ($\mathbf{W}_i^r$) has elements which are drawn from a normal distribution ($\mathcal{N}(0, 1.5)$). The reservoir has a connection matrix with values that are drawn from $\mathcal{N}(0, 1)$. Other properties of the reservoir are a spectral radius and a leak rate of 1. In this work the leak rate will be used to modulate the velocity of the generated trajectory. The connection of the bias to the readout layer is trained. The connection matrix $\mathbf{W}_b^r$ from (1) is drawn from a standard normal distribution and scaled by a bias-term of 0.5. The robot model is commanded by angular positions in degree. The network will however, explore this range (within the boundaries set by the

limiter) to find the correct angles. The introduced RLS-parameter α, defined in Section 2, is set to 1. The initial output weights $\mathbf{w}(0)$ are normalized random values ($\mathcal{N}(0,1)$).

4.2   Learning Kinematic Model

The desired spiral trajectory in task-space is similar to the one described in [7]. By connecting the proposed controller to the iCub simulation model, the controller will initially drive the robot arm randomly. Thanks to the RLS learning rule, fast adaptation of the output weights is achieved. As a results, fast learning of the kinematic model is acquired. After only 1000 time steps (time step = 270 ms) the robot starts following this spiral trajectory. Most feedback controllers use an error defined on the task-space to achieve the desired behavior. Although the proposed controller is not designed to minimize this error in task-space, it converges to the desired trajectory because the internal trained model corresponds to that of the iCub's arm. When such a model is achieved we could choose to continue the training online, learning the inverse kinematics in newly visited task-space regions. However, to evaluate the trained model at a certain point in time we will stop the training by setting $\Delta\mathbf{w} = \mathbf{0}$ in (5) and evaluate how well it continuous to follow a desired trajectory without learning. In Fig. 2(a) we show such generated trajectory of the iCub's arm (gray), which needs to follow a spiral trajectory similar (not the same) to the desired trajectory during learning. As demonstrated, the learned inverse kinematic model corresponds well with the iCub's arm for the desired end effector positions. However, because of the physical limitations of the robot some desired trajectory points, especially the ones closer to the robot ($Y_1 < -0.2$), are unreachable by the robot.

4.3   Generalization

Next, we investigate the transient and generalization behavior of the learned kinematic model. Instead of following a spiral trajectory we define some target corner points that form a specific shape (e.g.: a square or a star). So 4 target points for a square and 10 for a star. These points are all located on a plane perpendicular to the direction of the spiral. Each target point excites the network for 100 time steps. Afterwards, the next target point is given to the network. In other words, it is not necessary to follow a square or star trajectory, but the task is to reach the target corner points of each shape. The target points forming a square are located on the projection of the learned spiral data (dashed lines). As shown in Fig. 2(b) (bottom, left) the desired target points are reached. However, the generated movement between two different target points is demonstrating transient behavior, that is, it does not follow a straight line (shortest path between target points) but rather according to an arc (dark gray). The acquired kinematic model was learned by following a spiral trajectory, never reaching other regions of the task-space, which explains the transient arc behavior of the generated motion. The target points forming a star shape are, except for two, not located on the projection of the spiral trajectory. Although the learned kinematic model is based on the data seen while following the spiral trajectory, the

model is generalizing well to the other target points. Fig. 2(b) (bottom, right) shows small deviations in the reached target points (black dots), but it illustrates that the learned model generalizes well.

4.4 Modulation

The velocity at which each trajectory point is reached can be modulated after or while training. We achieve this by changing the leak rate γ of the reservoir states (1). As described before, changing $\gamma \in [0, 1]$ effectively changes the time scale of the system. Fig. 2(b) (top) demonstrates the effect of such modulation for multiple γ's after learning ($\Delta \mathbf{w} = \mathbf{0}$) and for different target points (e.g.: the target points forming the star shape). By decreasing γ, the distance between each sample point will decrease as well. As shown in the top plot of Fig. 2(b), the robot is unable to reach the target positions within 100 time steps when using $\gamma = 0.02$.

5 Conclusion

We presented a novel controller based on a recurrent neural network, which is able to learn the inverse kinematics of the iCub's robot arm fast, without any prior knowledge about the robot. By using an internal exploration approach, the proposed system starts learning a model of the arm. Although there is no error defined on the actual trajectory, the robot will eventually generate the desired motion. The system only uses observations of forward kinematic motion generation to learn a kinematic model of the robot arm. Consequently, the controller only observes a unique mapping of joint to task-space positions. Bad examples due to redundancy in the inverse kinematics, are not observed and thus eliminated in the actual control. Both generalization experiments demonstrate that when online learning is discontinued, the learned inverse kinematic model is restricted to the previously visited regions. Although generalizing well to unseen task-space regions, the claim of learning a full kinematic model is only valid when all possible end effector positions are visited during the online learning. From our experiments transient behavior of the network emerged as arced motions between the different presented target points. Finally, we demonstrated the velocity modulation of the controller and its effect on the resulting trajectory. In future work the applicability of this control approach to more advanced control tasks will be investigated and compared to classical control approaches.

References

1. Metta, G., Sandini, G., Vernon, D., Natale, L., Nori, F.: The iCub humanoid robot: an open platform for research in embodied cognition. In: Proc. of the 8th Workshop on Performance Metrics for Intelligent Systems, pp. 50–56. ACM (2008)
2. Tolani, D., Goswami, A., Badler, N.: Real-time inverse kinematics techniques for anthropomorphic limbs. In: Graphical Models, vol. 62, pp. 353–388. Elsevier (2000)
3. Grochow, K., Martin, S., Hertzmann, A., Popović, Z.: Style-based inverse kinematics. Transactions on Graphics (TOG) 23, 522–531 (2004)
4. D'Souza, A., Vijayakumar, S., Schaal, S.: Learning inverse kinematics. In: Intelligent Robots and Systems, vol. 1, pp. 298–303. IEEE (2001)
5. Peters, J., Schaal, S.: Learning operational space control. In: Proceedings of Robotics: Science and Systems (RSS). MIT Press, Philadelphia (2006)
6. Guez, A., Ahmad, Z.: Solution to the inverse kinematics problem in robotics by neural networks. In: International Joint Conference on Neural Networks, pp. 617–624. IEEE (1988)
7. Reinhart, R., Steil, J.: Reaching movement generation with a recurrent neural network based on learning inverse kinematics for the humanoid robot iCub. In: 9th IEEE-RAS International Conference on Humanoids, pp. 323–330. IEEE (2010)
8. Neumann, K., Rolf, M., Steil, J., Gienger, M.: Learning Inverse Kinematics for Pose-Constraint Bi-Manual Movements. From Animals to Animats 11, 478–488 (2010)
9. Jaeger, H., Lukosevicius, M., Popovici, D., Siewert, U.: Optimization and applications of echo state networks with leaky-integrator neurons. Neural Networks 20, 335–352 (2007)
10. Schrauwen, B., Verstraeten, D., Van Campenhout, J.: An overview of reservoir computing: theory, applications and implementations. In: Proc. of the 15th European Symposium on Artificial Neural Networks, D-side, pp. 471–482 (2007)
11. Sussillo, D., Abbott, L.: Generating coherent patterns of activity from chaotic neural networks. Neuron 63, 544–557 (2009)
12. Jaeger, H.: A method for supervised teaching of a recurrent artificial neural network. Patent Application, WO 2002/031764 A2 (2002)

Enhanced Codebook Model for Real-Time Background Subtraction

Munir Shah, Jeremiah Deng, and Brendon Woodford

Department of Information Science, University of Otago,
Dunedin, New Zealand
{mshah,ddeng,bjwoodford}@infoscience.otago.ac.nz
http://www.infoscience.otago.ac.nz/

Abstract. The CodeBook is one of the popular real-time background models for moving object detection in a video. However, for some of the complex scenes, it does not achieve satisfactory results due to the lack of an automatic parameters estimation mechanism. In this paper, we present an improved CodeBook model, which is robust in sudden illumination changes and quasi-periodic motions. The major contributions of the paper are a robust statistical parameter estimation method, a controlled adaptation procedure, a simple, but effective technique to suppress shadows and a novel block based approach to utilize the local spatial information. The proposed model was tested on numerous complex scenes and results shows a significant performance improvement over standard model.

Keywords: Codebook, Video Segmentation, Background Subtraction, Mixture of Gaussians, On line Learning, Online Clustering.

1 Introduction

Detection of moving (foreground) objects in a video is a fundamental task in most of the computer vision applications, such as video surveillance, activity recognition and human motion analysis in sports video sequences. This is a difficult task due to many challenges in realistic scenarios which include irregular motion in background,illumination changes, objects cast shadows, changes in scene geometery and presence of noise in training data, etc. [1,2,3].

To address the above-mentioned issues, a large number of research papers have been published in the previous decade [6,7,8,9,10,11,12,5]. Amonge these, the Mixture of Gaussians (MoG) exhibits a better ability to deal with gradual illumination changes due to better clustering capabilities. Since the introduction basic (MoG) [6] background model in 1999 numerous improvements [15,2,16] have been proposed. However, despite all of its enhancements its performance is still not satisfactory for some complex scenes especially those with quasi-periodic motion in the background and sudden illumination changes. Also, MoG is a parametric model and its parameters need to be hand-tuned for each particular scene, which makes it unsuitable for real-time applications [5,10]. Furthermore,

the temporal distribution of the pixels suggests that it is closer to the cylindrical model rather than Gaussian [5,10]. Hence, technically MoG is not a good choice for modeling in this case.

The CodeBook model [5,10] and its enhancements [10,1,17] perform better than other state-of-the-art methods both in-terms of segmentation accuracy and processing time. The cylindrical colour model and novel matching criteria are some of its distinguished features. However, according to the reported results [10,1,17] it does not achieve satisfactory results for some of the dynamic scenes like sudden illumination changes due to lack of automatic parameters selection mechanism. Thus, there is a need of further investigation in this regard to devise an automatic mechanism for optimal parameters selection. To the best of our knowledge no study exists on the investigation of how different configurations of the CodeBook parameters affect its performance.

In this paper we present several enhancements over basic CodeBook model which are, a robust statistical parameter estimation method, a controlled adaptation procedure, a simple, but effective technique to suppress shadows and a novel block based approach to utilize the local spatial information. The proposed model was tested on numerous complex scenes and result shows a significant performance improvement over standard model in terms of computational complexity, segmentation accuracy and memory efficiency.

The rest of the paper is organized as follows. In Section 2 we will review some related work, while Section 3 is an elaboration of the proposed approaches. The experimental results and analysis are presented in Section 4. Finally, we conclude this paper and propose possible future extensions in Section 5.

2   The CodeBook Model

In the basic Codebook background model, a Codebook (M) containing one or more codewords (c_i) is constructed for each pixel. Each codeword contains RGB vector $V_i = (R_i, G_i, B_i)$ and a six-tuple $aux_i = \{\check{I}, \hat{I}, f, \lambda, p, q\}$ where $\check{I}$ and $\hat{I}$ are the minimum and maximum brightness, f is the frequency with which it occurred, λ is longest interval of the time during which it is not re-occurred and p and q are the first and the last time when codeword was matched.

Let's suppose $X = \{x_1, x_2, x_3, ..., x_N\}$ is a training sequence containing N RGB vectors and $C = \{c_1, c_2, c_3, ..., c_L\}$ represents the codebook containing L codewords. In the training phase, try to find a matching codeword c_m based on color distortion and brightness bound as defined in [5]. If a match is found, update the matching codeword c_m using (1) and (2) as follows.

$$V_m = \alpha(X_t) + (1 - \alpha)V_m. \tag{1}$$

$$\sigma_m^2 = \alpha(\delta^2) + (1 - \alpha)\sigma_m^2. \tag{2}$$

Here, α is adaptation rate (usually between 0.0 and 1.0), δ is a color distortion [5], V_m is RGB intensity vector and σ_m^2 is global color distortion. If no matching

codeword is found create a new codeword and add it into M. The Codebook obtained in the previous step may become large due to noisy training sequence. Thus, at the end of initial training filter out codewords having large λ_i value.

The simple Codebook model is sufficient enough to model a scene where geometery of the scene does not changes over the time. However, in reality scenes may change after the training. For instance, a car parked in the parking starts moving, or sun comes out, etc. Therefore, to cope with these types of situations, a layered CodeBook model [5] was proposed by introducing an additional layer called the cache Codebook (H). In layered model, during the classification, if a matching codeword is found, classify this pixel as a background and update it using (1)and (2). Otherwise, classify it as foreground, create a new codeword, and place it in cache CodeBook H. Then move the codewords staying longer than some time in cache to the permanent codebook. After that, filter out both permanent and cache codebook (for details refer to [5]).

3   The Proposed Model

Although CodeBook has achieved better performance than other state-of-the-art models but still has unsatisfactory results for some complex scenes such as irregular background motion and sudden illumination changes. One of the major reasons leading to its degraded performance is its lack of automatic parameters selection mechanisim. The parameters controlling the behavior of this model are; Firstly, T_{del}, the thresholds used to filter out noisy codewords . Secondly, T_{add}, a threshold used for moving codewords from cache to main codebook. Thirdly, ϵ_1 and ϵ_2 are color distortion thresholds used in training and segmentation phase respectively. Fourthly, α, a learning rate. However, these pre-defined fixed values are not always adequate.

In-depth analysis of different scenes revealed that mostly background codewords have less variance than foreground codewords. As it can be seen from Fig.1, the first (background) codeword has much less variance compared to other foreground codewords. This motivated us to compute a variance σ_i^2 for each codeword and use this as a criterion to refine codebooks as follows.

$$\sigma_{m,t}^2 = \alpha(V_{m,t} - X_t)^2 + (1 - \alpha)\sigma_{m,t-1}^2. \tag{3}$$

Although using the variance as a criterion for refining CodeBook performs well for most of the cases, still there are some rare scenarios where background codewords may have high variance such as reflecting surfaces in the background or irregular motions. As it can be seen from the Fig.2, a pixel at the tip of the branch of the tree experienced an irregular motion due to which it has background codeword with high variance. Similarly, in some rare cases some foreground pixels may have low variance. For example, some smooth colored big object moving very slowly in video.To handle these cases we have introduced Min, Max bound along with the variance as follows.

$$\Theta^M(c_x) = \begin{cases} true, & \text{if } ((var_x > 2.5) \wedge ((p-q) > T_{del-min})) \vee ((p-q) > T_{del-max})) \\ false, & \text{otherwise} \end{cases} \tag{4}$$

	Codeword1			Codeword2			Codeword3			Codeword4		
	Y	Cr	Cb	Y	Cr	Cb	Y	Cr	Cb	Y	Cr	Cb
Variance	0.713	0.444	0.466	5.369	4.749	5.342	4.472	4.95	5.41	4.32	4.356	4.41
BoxMin	250	124	126	85	73	215	45	45	80	70	70	180
BoxMax	255	130	130	95	80	225	55	55	90	80	80	190

Frame 6   Frame 26   Frame 56   Frame 86

The graph shows the temporal distribution of this pixel.
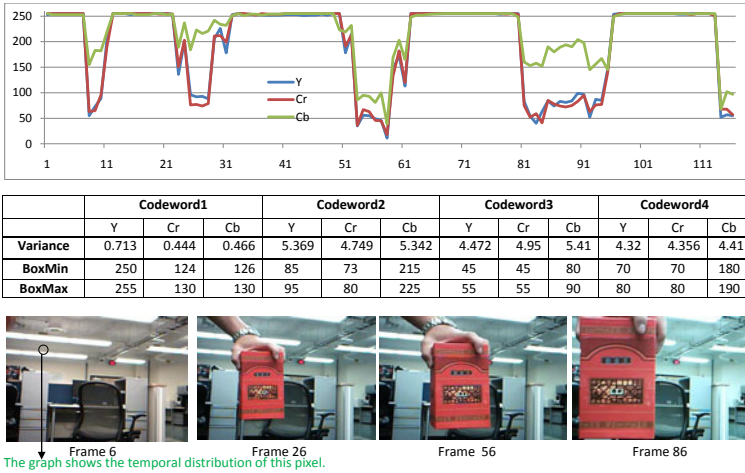
Fig. 1. Example demonstrating the usefulness of variance for differentiating between foreground and background codewords

Frame 1
The graph shows the temporal distribution of this pixel.

	Codeword1			Codeword2			Codeword3			Codeword4		
	Y	Cr	Cb	Y	Cr	Cb	Y	Cr	Cb	Y	Cr	Cb
Variance	5.044	4.116	4.979	141.5	121.5	122.3	6.652	2.743	3.469	4.811	2.445	4.587
BoxMin	210	114	128	4.877	4.026	4.442	121	115	129	149	114	121
BoxMax	227	121	142	136	120	119	144	122	140	172	120	138
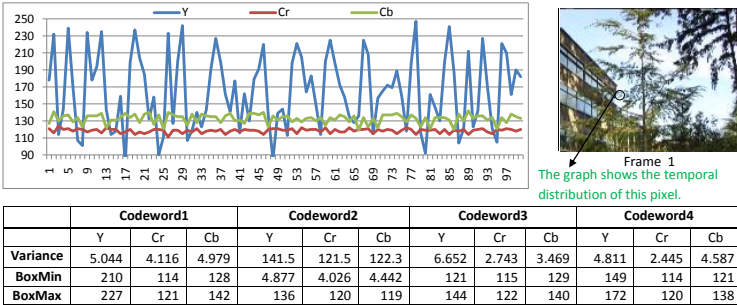
Fig. 2. Example demonstrating some rare cases where background codewords have high variance

$$\Theta^H(c_x) = \begin{cases} true, & \text{if } ((var_x > 2.5) \wedge ((p-q) > T_{H-min})) \vee ((p-q) > T_{H-max})) \\ false, & \text{otherwise} \end{cases} \quad (5)$$

$$\Phi(c_x) = \begin{cases} true, & \text{if } ((var_x \leq 2.5) \wedge (\lambda > T_{add-min})) \vee (\lambda > T_{add-max}) \\ false, & \text{otherwise} \end{cases} \quad (6)$$

Here, $\Theta^M(c_x)$ is a function for deleting codeword c_x from M, $\Theta^H(c_x)$ is a function for deleting codeword c_x from H, while $\Phi(c_x)$ is a function for moving codeword c_x from H to M. The max/min thresholds are pre-selected fixed values which keeps the model simple.

Furthermore, in original models codewords are adopted as shown in (1) and (2) and parameter α is controling adaptation speed, but finding an optimal α

value for particular scene is a hard task. In this regard, we have formulated an automatic and adaptive method for the selection of α for each particular scene. we observed that frequency of change can be used to find the dynamics of scene because complex scenes have high frequency of change, while on the other hand simple scenes have low frequency of change. Thus, we can model α using the fluctuation frequency as follows:

$$f_n = \begin{cases} 1, \text{if } \delta_n > \varphi \\ 0, \text{otherwise} \end{cases} \tag{7}$$

$$\mathrm{F}_{n,t} = (1 - \beta)\mathrm{F}_{n,t-1} + \beta f_n \tag{8}$$

$$\alpha_{n,t} = \begin{cases} 0.9\alpha_{n,t-1} + 0.1\frac{\mathrm{F}_{n,t}}{N}, \text{if } 0 \leq \alpha_{n,t-1} \leq 1, \\ \alpha_{n,t-1}, \qquad\qquad\quad \text{otherwise} \end{cases} \tag{9}$$

Here f is short-term, F is long-term frequency of pixels intensity fluctuation and δ_n is the absolute difference of pixel values in consective frames. Moreover, we have introduced controlled adaptation mechanism i.e. codewords will be adopted only when it is required as follows.

$$\mathrm{G}_c = \frac{\sum_1^N (\mathrm{F}_{N,t})}{N} \tag{10}$$

Here, F is long-term pixels frequency of change and N is normalization constant. If G_c is greater than threshold for certain period of time, it will be considered as complex scene and will be adopted accordingly.

In [18] a spatial CodeBook was proposed where the codebooks were constructed for individual pixels, but during the background segmentation phase instead of checking it in only its own codebook the pixel was also checked against

1: Initialize L to 0 and M^b to $\emptyset$.
2: **foreach** *Pixel in block b* **do.**
3: $X_t \leftarrow (R, G, B), I \leftarrow \sqrt{R^2 + G^2 + B^2}$.
4: Try to find a codeword c_m in M^b matching to X_t
5:   **if** Found **then**
6:     Update the codeword c_m using (1) and (2)
7:   **else**
8:     Increment L and create new codeword c_L by setting,
9:     $V_m \leftarrow (R, G, B)$ and $aux_L \leftarrow \langle I, I, 1, t-1, t, t \rangle$
10:    Add c_L in M^b
11:   **end if**
12: **end**
13: **foreach** *Codeword c_i in M^b* **do**
14:   Set $\lambda_i \leftarrow max\{\lambda_i, (N - q_i + p_i - 1)\}$
15: **end**

Algorithm 1. Block Based CodeBook Construction (Proposed)

1: During the training phase a codebook M^b was obtained by applying algorithm I.
2: Initialize L to 0 and H^b to $\emptyset$,
3: **foreach** *Pixel in block b* **do**
4: Try to find a matching codeword in M^b,
5: If match occurs with any of the codeword in M^b, classify this pixel as as background and update matching codeword by using (1) and (2).
6: Otherwise, classify it as foreground and Try to find a matching codeword in H^b. If found update it using (1) and (2). Otherwise create a new codeword and add it to H^b.
7: **end**
8: Clean H^b by using (5).
9: Move the cache codeword into M^b staying in H^b for longer than some time, using (6).
10: Clean M^b by using (4).
11: Repeat the process from step 4 for each incoming datum.

Algorithm 2. Block Based Layered CodeBook Model (Proposed)

the codebooks of neighboring pixels. In contrast, We have also introduced a novel block based CodeBook model in which codebook were constructed for 3X3 non-overlapping blocks instead of individual pixels. In this model each of the nine pixels in the block were used as training data instead of using global features for block. In this way many benefits can be achieved. Firstly, there will be fewer codewords which in return means less memory and less computation. Second, we are also able to exploit local spatial dependency between neighboring pixels for better segmentation result. The details are presented in Algorithm 1 and 2.

In Algorithm 2, during Step 7, each codeword in H is evaluated using (5), if $\Theta^H(c_x)$ is true codeword will be deleted. Also, in step 8 of Algorithm 2, each codeword in H is evaluated using (6), if $\Phi(c_x)$ is true codeword will be moved from H to M. Furthermore, in step 9, each codeword in M will be evaluated using (4), if $\Theta^M(c_x)$ is true codeword will be deleted from M.

4   Experimental Results and Analyses

For the validation of proposed model, we have selected Microsoft's Wallflowers data-set which contain several challenging benchmark indoor and outdoor scenes [19]. Furthermore, we have compared our model with state-of-the-art standard CodeBook (CB) [5] and Mixture of Gaussians (MoG) [6]. The parameters settings for MoG were $\alpha = 0.01$, $\rho = 0.001$, $K = 5$, $T = 0.8$ and $\lambda = 2.5\sigma$. Most of these parameters were taken from some of the previous studies [6,2,5,21,22]. For original CodeBook model, the authors in the original paper did not mention any specific value for the parameters, but in [21] the authors suggested some parameter values $\alpha = 0.01$, $\beta = 1.15$, $\epsilon = 10$ $T_{add} = 80$, $T_{del} = 100$, $T_H = 50$ and $T_M = (N/2)$. For the proposed models all the other parameters are same except $T_{add-min} = 25$, $T_{del-min} = 25$ and $T_{H-min} = 25$. In most of our experiments, unless otherwise stated, we have used same parameters setting all experiments.

We have used YCrCb color space instead of RGB color space, because during the experiments it has been found that some shadows can be suppressed by setting smaller threshold for Y component as compared to Cb and Cr. Therefore, for Y component we used threshold value $\epsilon=3$ instead of 10. More than 2-5% improvement in segmentation accuracy has been achieved by applying this simple, but effective heuristic.

Fig. 3. Segmentation results on Microsoft Wallflower data-set. The columns shows different data-sets while background modeling techniques are represented on rows.

Fig.3 and Fig.4 presents both qualitative and quantitative results (precision, recall and F-measures). As you can see in the Fig.3 and 4 all CodeBook family of background modeling techniques performed better than MoG for almost all the scenes. On the other hand, within the CodeBook family of background models, comparatively similar result has been achieved for WavingTrees, BootStrap and Camouflage scenes. Furthermore, our proposed models achieved significantly better results for all four types of different challenging scenes, whereas the standard CodeBook model failed in case of sudden illumination changes (LightSwitch).
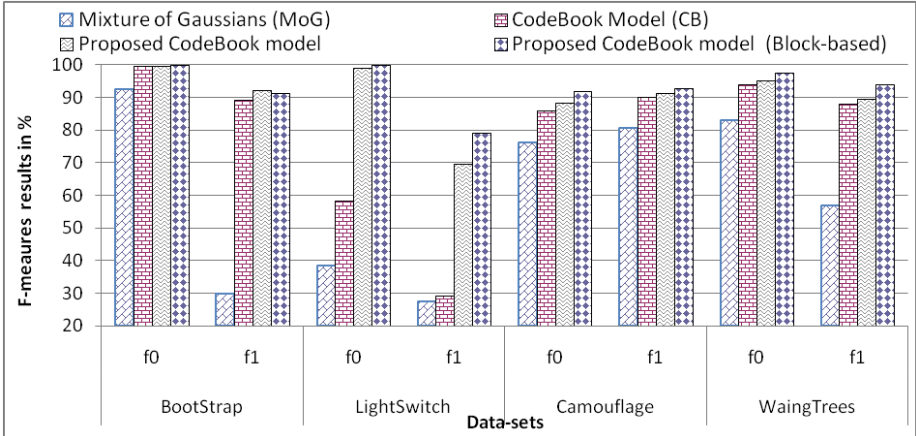
Fig. 4. Quantitative results for each model on different data-sets in the form of F-Measure F0 (background) and F1 (foreground)

The LightSwitch is comparatively stable scenes with certain sudden illumination changes. The standard CodeBook model fails for this data-set because it cannot differentiate between the stable and complex scenes. Our proposed models with a controlled adaptive learning mechanism seemed to have coped well with this type of situation. The proposed spatial model performed better than other models for some scenes, but slightly degraded performance has been experienced in case of BootStrap because pixels at the contours of the objects was wrongly classified due spatialy independency. Therefore, it is concluded that in the case of stable scenes spatial information is not always useful.

Although achieving limited improvement on segmentation accuracy for some video scenes, our proposed model performed significantly better than standard CodeBook model on speed and memory efficiency. Both implemented in C++ and running on a PC with a 2.26 GHz Quad core CPU, the standard CodeBook model goes as fast as 30 frames/sec, while the proposed model reaches 35 frames/sec. The size of the frames used for these experiments were s 320×240. The proposed models achieved a better segmentation accuracy and computational performance because of better adaptation mechanism which helps keep a very compact codebook.

5   Conclusion

In this paper we have presented an improved CodeBook background model for moving object segmentation in video by introducing several enhancements. With the introduction of simple but effective techniques significant improvements has been achieved in terms of segmentation accuracy, processing time and memory efficiency. The results are encouraging, but there are still some issues left unaddressed. For instance the color based CodeBook model cannot differentiate between background and foreground pixels if they both have similar colors. This will be one of our future directions.

References

1. Doshi, A., Trivedi, M.: Hybrid cone-cylinder codebook model for foreground detection with shadow and highlight suppression. In: IEEE International Conference on Video and Signal Based Surveillance, AVSS 2006, pp. 19–29 (2006)
2. Zivkovic, Z.: Improved adaptive gaussian mixture model for background subtraction. In: 17th International Conference on Pattern Recognition (ICPR 2004), vol. 2, pp. 28–31. IEEE Computer Society, Washington, DC, USA (2004)
3. Huang, T., Qiu, J., Ikenaga, T.: A foreground extraction algorithm based on adaptively adjusted gaussian mixture models. In: Fifth International Joint Conference on INC, IMS and IDC, pp. 1662–1667 (2009)
4. Darvish, P., Sills-Lavoie, M., Bilodeau, G.A.: A multiscale region-based motion detection and background subtraction algorithm. Sensors 10(2), 1041–1061 (2010)
5. Kim, K., Chalidabhongse, T.H., Harwood, D., Davis, L.: Real-time foreground-background segmentation using codebook model. J. Real-Time Imaging 11, 172–185 (2005)
6. Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. In: Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 246–252 (1999)
7. Cheng, J., Yang, J., Zhou, Y.: A novel adaptive gaussian mixture model for background subtraction. In: Proceeding of 2nd Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA 2005), Shanghai, China, pp. 587–593 (2005)
8. Friedman, N., Russel, S.: Image segmentation in a video sequences: A probabilistic approach. In: Proceeding of Thirteen Conference on Uncertainty in Artificial Intelligence, pp. 175–181 (1997)
9. Mahadevan, V., Vasconcelos, N.: Background subtraction in highly dynamic scenes. In: Proceeding of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2008), pp. 1–6 (2008)
10. Kim, H., et al.: Background subtraction using generalized gaussian family model. IEEE Electronics Letters 24(5), 473–482 (2006)
11. Cheng, J., Yang, J., Zhou, Y., Cui, Y.: Flexible background mixture models for foreground segmentation. J. Image and Vision Computing 24(5), 473–482 (2006)
12. Zhong, J., Sclaroff, S.: Segmenting foreground objects from a dynamic textured background via a robust kalman filter. In: Proceeding of Ninth IEEE International Conference on Computer Vision (ICCV 2003), Washington, DC, USA, p. 44 (2003)
13. White, B., Shah, M.: Automatically tunings background subtraction parameters using particle swarm optimization. In: Proceeding of IEEE International Conference on Multimedia and Expo. (2005)

14. Figueiredo, M.A.T., Jain, A.K.: Unsupervised learning of finite mixture models. J. IEEE Transaction on Pattern Analysis and Machine Intelligence 24(3), 381–396 (2000)
15. Herrero, S., Bescos, J.: Background Subtraction Techniques: Systematic Evaluation and Comparative Analysis. In: Blanc-Talon, J., Philips, W., Popescu, D., Scheunders, P. (eds.) ACIVS 2009. LNCS, vol. 5807, pp. 33–42. Springer, Heidelberg (2009)
16. Lee, D.-S.: Effective gaussian mixture learning for video background subtraction. J. IEEE Transactions on Pattern Analysis and Machine Intelligence 27, 827–832 (2005)
17. Guo, J.-M., Hsu, C.-S.: Cascaded background subtraction using block-based and pixel-based codebooks. In: Proceedings of the 2010 20th International Conference on Pattern Recognition, ICPR 2010, pp. 1373–1376 (2010)
18. Wu, M., Peng, X.: Spatio-temporal context for codebook-based dynamic background subtraction. AEU - International Journal of Electronics and Communications 64(8), 739–747 (2010)
19. Kentaro, T., John, K., Barry, B., Brian, M.: Wallflower: Principles and practice of background maintenance. In: IEEE International Conference on Computer Vision (CCV 2009), p. 255 (1999)
20. Piccardi, M.: Background subtraction techniques: a review. In: IEEE International Conference on Systems, Man and Cybernetics, vol. 4, pp. 3099–3104 (2004)
21. Wu, Y., Zeng, D., Li, H.: Layered video objects detection based on lbp and codebook. In: First International Workshop on Education Technology and Computer Science, vol. 1, pp. 207–213 (2009)
22. Li, Y., Chen, F., Xu, W., Du, Y.: Gaussian-Based Codebook Model for Video Background Subtraction. In: Jiao, L., Wang, L., Gao, X.-b., Liu, J., Wu, F. (eds.) ICNC 2006. LNCS, vol. 4222, pp. 762–765. Springer, Heidelberg (2006)

Color Image Segmentation Based on Blocks Clustering and Region Growing

Haifeng Sima[1,2], Lixiong Liu[1,*], and Ping Guo[1,*]

[1] School of Computer Science and Technology,
Beijing Institute of Technology, Beijing 100081, China
[2] Department of Computer Science and Technology
Henan Polytechnic University, Jiao zuo, 454003, China
lxliu@bit.edu.cn, pguo@ieee.org

Abstract. In order to overcome the discontinuity in clustering segmentation, a novel color image segmentation algorithm is proposed, which is based on seeds clustering and can locate the seeds of regions quickly. Firstly, the image is divided into a series of non-overlapping blocks with the size of $n{\times}n$ pixels in HSI color space. For each block, the centroid pixel of salient homogeneous region is selected as a feature point of the block. Secondly, based on the principles of color similarity centroids are clustered to obtain the clustered centroids as seeds for region growing. Finally, invalid and noisy regions are merged to get the complete segmentation results. Comparing with other segmentation algorithms, the experimental results demonstrate that the proposed method can accurately segment regions and objects, it outperforms other methods in terms of human visual perception.

Keywords: color image, image segmentation; seeds clustering; region growing; region merging.

1 Introduction

In computer vision, image segmentation is the basis of the image analysis and understanding. It is widely used in image retrieval, medical image analysis, target location and other fields. Extraction of high-level semantics of images requires obtaining accurate low-level segmentation features, so obtaining clear and accurate outline of goals and objects in the image is important. According to the theoretical basis for segmentation, existing color image segmentation methods can be divided into the following types: Region-based methods, boundary detection, thresholding and clustering [1-5]. Each method has its advantage for certain images or applications. Recently, most popular image segmentation methods integrate multiple technologies to achieve satisfactory results [6, 7].

The region growing method is relatively stable for segmentation and can meet with the needs for many real applications. Many region growing algorithms derive from

---

* Corresponding authors.

the classic algorithms SRG (Seed Region Grow) [8]. It selects seed points in gray level image, and the algorithms will generate different results with the different growing order. A. Mehnert improved SRG by taking parallel computing approach, to eliminate the dependency of growing order [9]. J. Fan *et al* [10] proposed a hybrid method of edge extraction and seed growth. It detects the boundary in the YUV color space, and chooses the regional centroids of boundary as the seed. Shih. F. Y. *et al* [11] merge adjacent pixels based on pixel similarity in YCrBr space, obtain many different pixels sets as initial seeds, and merge regions based on adjacent and homogeneous after growing. Ye *et al* [12] take incremental growing model to form the initial image partition by connectivity of similar color pixels, and automatic merge the initial partitions according to color and spatial information. In addition to regional growing algorithm, most popular image segmentation methods integrate multiple technologies in order to achieve satisfactory results. In JSEG algorithm [13], image was segmented by preliminary multi-scale clustering, then grows regions by selected seed pixels. The segmentation result is influenced by the scale and complicated process.

To integrate the advantages of clustering and region growing, we propose a color image segmentation algorithm in HSI space. In this algorithm, an image is divided into non-overlapping mini-blocks with the size of $n \times n$ at first. In each block, the centroid of homogeneous region is extracted as a sub-block feature. Then centroids are clustered based on the principles of color similarity and clustering centroid sets are obtained as seeds for region growing. Finally, invalid and similar regions are merged to obtain segmentation results. This proposed algorithm is called as BCRG (blocks clustering and region growing). Compared with other segmentation algorithms, segmentation results demonstrate that the proposed algorithm has the good performance on getting accurate regions in accordance with the results of visual perception. The outline of the proposed algorithm is shown in Fig.1.
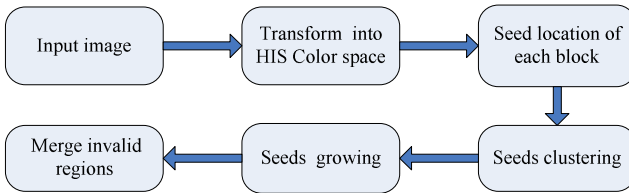
Fig. 1. Outline of the image segmentation algorithm

2   Image Segmentation Algorithm Based on Seeds Clustering

2.1   Selected Seeds of Blocks

For an image, the color difference is the key factor that brings the visual difference between different objects. In order to localize the seeds of image regions accurately, let Hue component be the main basis for seed selection of regions. Firstly, the image is divided into a number of $n \times n$ pixels independent regional blocks. The represent pixels are selected as the seeds from blocks for future growth:

(1) Compute the statistic   pixels distribution of Hue in a selected block;
(2) Calculate pixels number $N[p]$ according peak values of hue in a block; choose interval $(-\delta < H_{max}(p) < \delta)$ in which it contains maximum pixels H_δ;
(3) Define pixels H_δ as homogeneous pixels whose hue value in the interval in the block, homogeneous pixels assemble a small region R represented the block;
(4) Obtained the centroid of the region R as the color of feature points of given block;

All centroids obtained in this process represent the color feature of each block, which are the seeds for region growing in next step. The centroid located approach of blocks is illustrated in Fig.2.
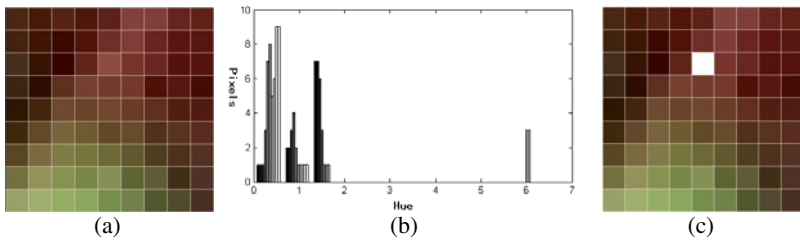
Fig. 2. Display feature point selection of blocks. (a) a selected block, (b) the statistics color distribution of all pixels in the block, (c) white pixels show the centroid of the given block.

2.2   Seeds Dynamic Clustering

Regional seed pixels should satisfy two constraints [14]. First, the pixels should be similarity with neighboring pixels. Second, the seeds of different regions are not adjacent. The centroids satisfy the first constrains. Next, clustering centroids accordance can satisfy the second constraint. All centroids are aggregate into the seeds of different regions. In order to achieve better segmentation, we proposed a dynamic division method. The seeds are divided into 12 intervals by Hue circle, which is shown in Fig.3. And the cluster number is adjusted dynamically during the clustering process.

Fig. 3. Division of Hue space

Based on the hue division, seed pixels of the image are classified into 12 intervals as initial cluster. The introduction of *a priori* knowledge to clustering can speed up

the convergence. Here pixel feature vector is defined as $\{H_j, S_j, I_j\}$. The measurement of the similarity between vectors can be calculated by:

$$Dist = \sqrt{(I_i - I_j)^2 + (S_i - S_j)^2 - 2 S_i S_j (H_i - H_j)^2} \ . \tag{1}$$

Clustering process is described as follows:

Input: data vectors of block centroids sets, and 12 initial classes $C(k)$.
Output: the final class number and classification label of pixels.

(1) Compute the center of each class C_k, j=1, 2,…, k, composed cluster center set: $C(k)$, initial value k=12.
(2) Compute the maximum distances $D_{max}^{C_k}$ of inner class between all pixels and C_k and minimum distances $D_{min}^{p,q}$ between two classes C_p and C_q by equation (1).
(3) If $D_{min}^{p,q} < \overline{D_{max}}$, merge the two classes C_p and C_q; if $D_{max}^{j} > \overline{D_{min}}$, split the class C_j into two classes; where $\overline{D_{max}}$ represents average value of $D_{max}^{C_k}$ and $\overline{D_{min}}$ represents average value of $D_{min}^{p,q}$;
(4) Adjust the class centers with equation (2), where x represent any pixels in the new cluster. Update new centers of the classes sets C_k, and return to step (2)

$$C_j = \frac{1}{N} \sum_{x \in C_{new}} x(H, I, S) \cdot \tag{2}$$

(5) Clustering process is terminated until clustering results never change.

The seeds selection after dividing the image into blocks brings two benefits: (1) it will reduce the complexity of seeds selection, so post-growing process can be implemented by parallel; (2) reduce the local impact of noise on the zoning, and decrease the appearance of small disconnected area. The centroids and clustering results are shown in Fig.4.

Fig. 4. Seeds clustering illustration, (a) peppers, (b) centroids (c) clustering of centroids

2.3   Rules for the Region Growing

Region growing is based on the similarity between neighbor pixel and overall region. New pixels are compared with regional average similarity while merging pixels. So a set of variables is defined to store regional average feature, in which it contains hue,

brightness and saturation component. 8 connected constraints are taken to guide the growth of new pixels: If the similarity of an expected pixel in eight-neighbor is higher than given threshold, it is identified belonging to the region. When pixel was merged in, the records of the area size will be update: the variable *reg_size* plus 1. New pixels are continuing merged until there are no satisfied pixels. The regional feature values updated as follows:

$$\overline{H}' = \frac{(Hnew + \overline{H})}{reg\_size + 1}, \overline{S}' = \frac{(Snew + \overline{S})}{reg\_size + 1}, \overline{I}' = \frac{(Inew + \overline{I})}{reg\_size + 1}. \tag{3}$$

The similarity between the pixels during region growing is calculated by follows:

$$S_i = \exp[\cos(H_j - \overline{H}) + \frac{(S_i - \overline{S})^2 + (I_i - \overline{I})^2}{\sigma_s^2 + \sigma_i^2}], \tag{4}$$

$$\sigma = \sqrt{\frac{1}{N_i} \sum_{seed(i)} (\Gamma_i - \Lambda)^2} \quad \Gamma = \{S, I\}, \Lambda = \{\overline{S}, \overline{I}\}. \tag{5}$$

In which σ_s and σ_i represent variance of intensity and saturation components in *Seed(i)*, respectively, when using Gaussian function computes similarity to the normalized variables. The maximum similarity within the class is taken as threshold calculated as formula (6).

$$S_i^t = Max(S_t) \quad t \in seed(i). \tag{6}$$

The descriptions of growing approach are as follows:

(1) Select the pixels set *seed* (*i*) for the growth;
(2) Calculate similarity of neighborhood pixels, if $S_{[j]} < S_i^t$, merge pixel (*j*) in to Region (*i*), until no new pixel satisfy the constraints;
(3) Classified pixels with no minimum in accordance with the principle of similarity classes are merged into adjacent cluster;
(4) When all pixels are classified terminate running algorithm, otherwise return to (2).

Fig. 5. The illustration of region merging

2.4   Merging Unnecessary Regions

Clustering process will generate some small sets of isolated pixels and produce corresponding regions during growing. These small regions located in the outline of other regions are unnecessary for the segmentation. Adhere to the two principle of the

process of merging, space adjacent and color similar, these regions can be eliminated. First calculate the closed regions, if the color difference between two areas is less than a particular value, then merge them; Second, check all the number of pixels within a region, if the number of pixels in the proportion is less than a certain percentage in the image, merge the region to adjacent regions with the highest similarity. The segmentation and image after region merged are shown in Fig.5.

3   Experimental Analysis

In order to verify the effectiveness of the proposed algorithm, we select 50 color images from the experimental database with uniform resolution 384×256 pixels. These images include different types of images, such as landscapes, animals, people and architecture with different scale of goals in the images. The Block-scale parameter is set to $n = 9$. In Fig. 6, it shows 8 typical images from test results of segmentation.

The time complexity of the algorithm can be analyzed on considering two parts of the seed positioning and growing. The time complexity of computing the block centers is $O(n)$, complexity of iteration for locating cluster centers is $O(n\log(n))$, the time complexity of growing is $O(n)$, the overall time complexity is approximately $O (n + n\log (n))$. Compared with previous algorithms, this seed selection algorithm has two advantages: (1) using blocks feature for dynamic clustering can obtain more reasonable region than clustering pixels; (2) the seed is the pixel group gathered from represent region, pixel group of more than a single pixel regional representation can speed up the growth compared the pixel-based representation.

Fig. 6. Some segmentation results from test database

Here, for comparison, we select four different types of segmentation algorithms, including the classic normalization cuts (N-Cuts) algorithm [14], JSEG segmentation algorithm [13], the active contour algorithm (ACM) [15] and regional boundaries algorithm (LESBR) [16]. In Fig. 7, it shows the segmentation results of five algorithms with three type images (flowers, landscape, and horse). The lines in images represent the segmentation results by an algorithm. The segmentation results shown in Fig.7 from first row to the fifth row are our proposed BCRG algorithm, normalized cut algorithm, JSEG algorithm, ACM algorithm and LESBR segmentation results, respectively. From Fig. 7, it can be seen that the proposed BCRG algorithm shows an ideal segmentation. Comparing with other segmentation algorithms, the experimental results demonstrate that the BCRG algorithm can segment regions and objects accurately, it outperforms other methods in terms of human visual perception.
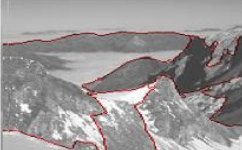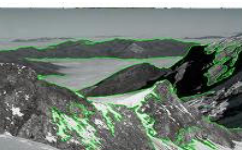
Algorithm	flowers	landscape	horse
BCRG			
NC			
JSEG			
ACM			
LESBR			

Fig. 7. Comparison of 5 segmentation algorithms with flower, landscape and horse image

4   Conclusions

In this paper, a novel color image segmentation algorithm based on seeds clustering is proposed, which combines advantages of region growing and clustering, adopts dynamic clustering of image block feature for seed position, grows and merges in HSI space. Experimental results show that the algorithm has good performance in different types of color image. The segmentation regions meet the requirements of human visual perception. Comparing with the classical region growing segmentation algorithm, BCRG algorithm can obtain the target and background accurately in the different types of images. In the future work, we will focus on introducing complex textures into the algorithm in order to improve the accuracy and applicability of segmentation algorithms.

References

1. Prados-Sure, B., Chamorro-Martinez, J., Sanchez, D., Abad, J.: Region–based fit of color homogeneity measures for fuzzy image segmentation. Fuzzy Sets and Systems 158, 215–229 (2007)
2. Zollerand, T., Buhmann, J.M.: Robust Image Segmentation Using Resampling and Shape Constraints. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(7), 1147–1164 (2007)
3. Aghbari, Z.A., RubaAl-Haj: Hill-manipulation: An effective algorithm for color image segmentation. Image and Vision Computing 24, 894–903 (2006)
4. Shih, F.Y., Cheng, S.: Adaptive mathematical morphology for edge linking. Information Sciences 16(1-4), 9–21 (2004)
5. Nicolas, V.B., Zludovic, B., Postaire, J.G.: Color image segmentation by pixel classification in an adapted hybrid color space application to soccer image analysis. Computer Vision and Image Understanding 90, 190–216 (2003)
6. Chen, T.W., Chen, Y.L., Chien, S.Y.: Fast image segmentation based on K-Means clustering with histograms in HSV color space. In: 2008 IEEE 10th Workshop on Multimedia Signal Processing, pp. 322–325 (2008)
7. Park, J.H., Lee, G.S., Park, S.Y.: Color image segmentation using adaptive mean shift and statistical model-based methods. Computers and Mathematics with Applications 57, 970–980 (2009)
8. Adams, R., Bischof, L.: Seeded region growing. IEEE Transactions on Pattern Analysis and Machine Intelligence 16(6), 641–647 (1994)
9. Mehnert, A., Jackway, P.: An improved seeded region growing algorithm. Pattern Recognition Letters 18, 1065–1071 (1997)
10. Fan, J.P., Zeng, G., Body, M., Hacid, M.S.: Seeded region rowing: an extensive and comparative study. Pattern Recognition Letter 26(8), 1139–1156 (2005)
11. Shih, F.Y., Cheng, S.X.: Automatic Seeded Region Growing for Color Image Segmentation. Image and Vision Computing 23(10), 877–886 (2005)
12. Ye, Q.X., Gao, W., Wang, W.Q., Huang, T.J.: A Color Image Segmentation Algorithm Using Color and Spatial Information. Journal of software 154, 522–530 (2004)
13. Deng, Y., Manjunath, B.S.: Unsupervised segmentation of color-texture regions in images and video. IEEE Transactions of Pattern Analysis and Machine Intelligence 23(8), 800–810 (2001)
14. Shih, J.B., Malik, J.: Normalized Cuts and Image Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(8), 888–905 (2000)
15. Zhang, K.H., Zhang, L., Song, H.H., Zhou, W.: Active contours with selective local or global segmentation: a new variational approach and level set method. Image and Vision Computing 28(4), 668–676 (2010)
16. Donoser, M., Riemenschneider, H., Bischof, H.: Linked Edges as Stable Region Boundaries. In: Proceeding of Conference on Computer Vision and Pattern Recognition, pp. 1665–1672 (2010)

Speed Up Spatial Pyramid Matching Using Sparse Coding with Affinity Propagation Algorithm

Rukun Hu and Ping Guo⋆

Image Processing and Pattern Recognition Laboratory
Beijing Normal University, Beijing 100875, China
`pguo@ieee.org`

Abstract. Recently support vector machines (SVMs) combining spatial pyramid matching (SPM) kernel have been highly successful in image annotation. And linear spatial pyramid matching using sparse coding (Sc-SPM) scheme was proposed to enhance the performance of SPM both in time and annotation accuracy. However, both of these algorithms suffer from expansibility problem, and ScSPM needs quite a long time for codebook construction. In this paper, we proposed an adjusted framework for the ScSPM algorithm, which applies multi-level affinity propagation (AP) algorithm to the codebook construction process (AP-ScSPM). This novel approach can remarkably reduces the time complexity of codebook construction process. Furthermore, as AP algorithm can automatically determine the representative vector number, the expansibility of the algorithm is improved. By a series of experiments, we find that the proposed framework greatly reduces the time of codebook construction process and has the same performance in terms of annotation accuracy with ScSPM.

Keywords: Image annotation, Affinity propagation, Spatial pyramid matching, Scale invariant feature transform.

1 Introduction

In recent years, contend-based image annotation has been the subject of a significant amount of research. Various methods have been proposed [1][2][3][4][5], and the bag-of-features (BoF) [6] model has been extremely popular in image annotation. The model treats an image as a collection of unordered appearance descriptors extracted from local patches, quantizes them into discrete "visual words", and then computes a compact histogram representation for semantic image classification, e.g. object recognition or scene categorization [6].

As the BoF approach discards the spatial information of local descriptors, the descriptors' representation power is limited. Based on the BoF model, an extension called spatial pyramid matching (SPM) [1] was proposed and has made a remarkable success on a range of image classification benchmarks like Caltech-101 and Caltech-256, and was the major component of the state-of-the-art systems [6]. However, as the SPM model use the nonlinear SVM as classifier, great

---

⋆ Corresponding author.

computational and memory complexity in the training phase need to be paid, which will limit the application of this model in real-world. To solve this problem, an extension of the SPM approach, named Linear Spatial Pyramid Matching Using Sparse Coding (ScSPM) [6] was proposed.

In the ScSPM, a spatial-pyramid image representation based on sparse coding (SC) of scale invariant feature transform (SIFT) features, instead of the $k-$means vector quantization (VQ) in the traditional SPM was proposed. The ScSPM approach is naturally derived by relaxing the restrictive cardinality constraint of VQ. Furthermore, unlike the original SPM that performs spatial pooling by computing histograms, the ScSPM method uses max spatial pooling which is more robust to local spatial translations and more biological plausible [6]. As to the classifier, the ScSPM use simple linear SVMs instead of nonlinear ones which dramatically reduces the training complexity to $\mathcal{O}(n)$, and obtains a constant complexity in testing, while still achieving an even better classification accuracy in comparison with the traditional nonlinear SPM approach [6]. But in the ScSPM algorithm, there are still two disadvantages: 1) The codebook construction process is time consuming; 2) The codebook size is empirically pre-settled, and the expansibility of the ScSPM algorithm is limited.

In this work, multi-level AP algorithm is applied to train the codebook used in ScSPM algorithm. The performance of the ScSPM and the AP-ScSPM is studied in this paper. In Section 2 the feature used in the work is discussed and the codebook construction technique is presented in details. The experiment details are given in section 3. Finally, conclusion is given in Section 4.

2   Feature Extraction

According to reference [1], all images can be represented by a bag of feature vectors. And if a codebook can be used to classify those feature vectors, the image can be then presented by a single histogram vector. In [1], $k-$Mean algorithm was used to train the codebook, and in [6], sparse coding algorithm and spatial pooling methods are chosen. In this work, AP algorithm is used to train the codebook, and the spatial pooling idea is kept when coding the SIFT features (the same as that in [6]). The advantage of this framework will be discussed in section 3.

2.1   Multi-level AP Framework for Codebook Construction

As discussed above, the codebook plays a very important role in the feature mapping process. In our study, we find that cluster centers can be used for codebook construction. And with the application of AP algorithm [8], the codebook construction process is faster than that in [6].

What is more, compared with other clustering method, AP algorithm is robust to the initial value, it can automatically determine the cluster number based on the data's distribution feature, and also can adjust the cluster result based on the user's interest by tuning the preference parameter P of the algorithm [8][9].

AP algorithm is based on graph theory, and the time complexity increases rapidly with the raise of the number of data. One possible solution for this problem when dealing with large data set is to divide the data set into small subsets and hieratically apply AP algorithm to pick the representative data in each subset. In this work, both two-level and three-level AP frameworks (as
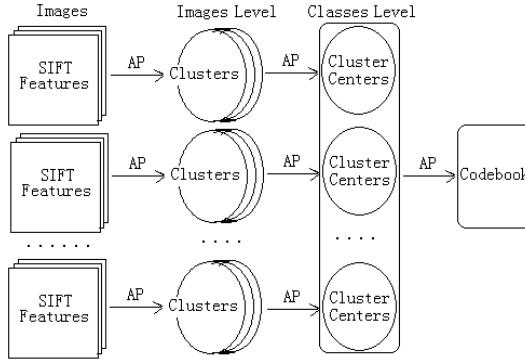
Fig. 1. Multi-level AP cluster framework for codebook construction

shown in Fig. 1) for codebook construction are studied and the results are shown and analyzed in section 3.

In Fig. 1, first, AP is used to select feature vectors from each training image; then apply AP to select representative vectors for each class. In the two-level framework, those selected vectors from each class are put together to form the codebook. As for the three-level AP framework, another AP algorithm is added to compress the codebook size. According to the experiment results, those two framework can remarkably reduce the time complexity of codebook construction process with good annotation accuracy.

3   Experiment

To investigate the performance of the proposed framework for codebook construction used in ScSPM, we do experiments on two image datasets and study the performance of both ScSPM and AP-ScSPM.

3.1   Summary of Experiments

In the experiment, two image datasets are used: thirteen scene categories and Caltech-101. For the thirteen scene categories, 13 classes with 2926 images are used. And for the Caltech-101 dataset, 40 classes with 2643 images are selected. Furthermore, the 40 classes are randomly divided into two subsets with 20 classes each. In the following part, we refer these three datasets as Caltech-101-40, Caltech-101-20-a and Caltech-101-20-b. In all those datasets , most images are medium resolution , i.e. about 300×300 pixels.

For the ScSPM, 200,000 patches are randomly chosen for codebook construction, and the codebook size is set to 1024. For the AP-ScSPM, different number of images, 25 and 30 are tried in the experiment, for each class are investigated to train the codebook. And both two-level and three-level AP framework are studied. For these two algorithms, 25 or 30 images are randomly chosen from each class, respectively, for SVM training and the rest for testing.

In the following part, both the algorithm time complexity and expansibility are recorded and analyzed. In the tables, the time is recorded in H (hour): m (minute): s (second) format and the annotation accuracy is an average value over five runs. The label "30-LVL2" means 30 images are randomly selected for codebook construction in AP-ScSPM, and a two-level AP framework is used, and so on. The record in table 2 and 5 is from experiments run on one computer, the rest are from experiments run on another one.

3.2   Algorithm Description

In this section, the proposed multi-level AP framework for codebook construction process is presented in details.

Consider an image database $\mathcal{T} = \{I_1, I_2, ..., I_N\}$ and a semantic label vocabulary $\mathcal{L} = \{w_1, w_2, ..., w_N\}$. In the training phrase, a training set $\mathcal{D}=\{(I_1, W_1), ..., (I_N, W_N)\}$ of image-caption pairs is assumed, where $I_i \in \mathcal{T}$ and $W_i \subset \mathcal{L}$.

  i. For each semantic class $w_i \in \mathcal{L}$, randomly select n_i training images as subset ws_i.
 ii. For each image in WS=$\{ws_1, ws_2,..., ws_N\}$,split it into overlapping patches with the size 16×16, and get a SIFT descriptor from each patch.
iii. For SIFT feature vectors from each image, apply AP algorithm to select representative vectors VI_i.
 iv. For the training vectors of each class $VT_i=\{VI_1, VI_2,..., VI_{n_i}\}$, pick the cluster centers VC_i with AP algorithm.
  v. In the two-level AP framework, put together all the cluster centers of each class to construct the codebook V=$\{VC_1, VC_2,..., VC_N\}$. As for the three-level framework, AP algorithm is applied to V, and a codebook with smaller size is generated.

3.3   Time Cost and Annotation Accuracy Analysis

In this work, the performance of the ScSPM and the AP-ScSPM are investigated. The performance of AP-ScSPM using different image numbers for traing and different levels of AP algorithm for vector quantization on dataset Caltech-101-20-a is recorded in table 1. As we can see that in AP-ScSPM, very little images are required for the codebook construction (about 25 images for each class). And in the three-level AP framework, the codebook size is reduced to a very small level. All the unimportant vectors are cut away, the dimension of the histogram vector for each image is largely reduced. As a result, less time is needed for SVM training. And at the same time, the average accuracy for image annotation of the AP-ScSPM algorithm stay at a high level.

Table 1. Comparison annotation performance using different image numbers and different levels of AP algorithm for codebook construction on Caltech-101-20-a dataset

Variable	30-LVL2	30-LVL3	25-LVL2	25-LVL3
Codebook Construction Time	68m4s	74m26s	56m26s	59m51s
Codebook Size	128×1562	128×191	128×1426	128×162
Sparse Coding Time	39m38s	23m11s	35m9s	21m25s
ScSIFT Dimension	1×32802	1×4011	1×29946	1×3402
SVM Training Time	2m44s	28s	2m41s	23s
SVM Testing Time	1s	1s	1s	1s
Average Accuracy	0.843894	0.820849	0.843091	0.817134
Standard Deviation	0.013249	0.056145	0.013011	0.00494

Table 2. Comparison annotation performance using different framework on SceneClass-13 image dataset

Variable	ScSPM	AP-ScSPM-LVL2	AP-ScSPM-LVL3
Codebook Construction Time	27h1m4s	3h6m26s	5h43m22s
Codebook Size	128×1024	128×1706	128×129
Sparse Coding Time	3h20m38s	3h52m9s	2h24m
ScSIFT Dimension	1×21504	1×35826	1×2901
SVM Training Time	1m9s	3m44s	10s
SVM Testing Time	1s	1s	1s
Average Accuracy	0.794337	0.798154	0.756115
Standard Deviation	0.011569	0.013385	0.006178

From table 1 to table 5, the performance for both ScSPM and AP-ScSPM on different datasets is recorded. From the tables we can see that, on all datasets tried in the experiments, the codebook construction time of AP-ScSPM is far more shorter than that of SPM and both algorithm can achieve high annotation accuracy. Meanwhile, the three-level AP-ScSPM algorithm can reduce the complexity of the codebook remarkably and SVM training time with very little cost of annotation accuracy.

In the ScSPM codebook training process, the codebook is initialized by $k-$Means algorithm which is sensitive to the initial value and unstable. So iteration process is necessary for a stable codebook, which will take a long time. For example, in the code of [6], 50 cycles are used and this will take about 16 hours to obtain a stable codebook using dataset Caltech-101-20-a. However, if multi-level AP algorithm is used, the construction time will be reduced to about 1 hour with only a tiny drop in annotation accuracy.

3.4   Algorithm Expansibility

As discussed in [6], the codebook size in ScSPM is experientially settled by the author. As the data increase, this value must be experientially resettled and the codebook must be construction again, which seriously limit the expansibility of

Table 3. Comparison annotation performance using different framework on Caltech-101-20-a image dataset

Variable	ScSPM	AP-ScSPM-LVL3
Codebook Construction Time	16h34m27s	59m51s
Codebook Size	128×1024	128×162
Sparse Coding Time	30m38s	21m25s
ScSIFT Dimension	1×21504	1×3402
SVM Training Time	1m56s	23s
SVM Testing Time	1s	1s
Average Accuracy	0.845240	0.817134
Standard Deviation	0.012530	0.00494

Table 4. Comparison annotation performance using different framework on Caltech-101-20-b image dataset

Variable	ScSPM	AP-ScSPM-LVL3
Codebook Construction Time	18h26m32s	1h15m28s
Codebook Size	128×1024	128×240
Sparse Coding Time	53m57s	33m28s
ScSIFT Dimension	1×21504	1×5040
SVM Training Time	2m13s	32s
SVM Testing Time	1s	1s
Average Accuracy	0.746360	0.732775
Standard Deviation	0.011988	0.019220

the algorithm. However, with the application of AP algorithm, the codebook size can be automatically decided by the algorithm itself based on the data feature as shown in the tables bellow.

With the increase of class number, the size of the codebook become larger and larger. Though this makes sense, the large size codebook will significantly slow down the sparse coding, SVM training and testing process. One possible solution for this problem is to use the three level AP framework to train the codebook.

Table 5. Comparison annotation performance using different framework on Caltech-101-40 image dataset

Variable	ScSPM	AP-ScSPM
Codebook Construction Time	14h37m5s	6h3m24s
Codebook Size	128×1024	128×388
Sparse Coding Time	4h52mm15s	3h35m27s
ScSIFT Dimension	1×21504	1×8148
SVM Training Time	16m15s	3m23s
SVM Testing Time	1s	1s
Average Accuracy	0.757612	0.738709
Standard Deviation	0.015682	0.010446

From records in the table, it can be found that the three level framework can efficiently shrink the codebook size with little cost of annotation accuracy. According to the experiment, with the three level AP framework, the codebook size is about ten times the number of classes.

4 Conclusion

This paper studies the performance of ScSPM algorithm and proposed an improved codebook construction process technique based on AP algorithm for ScSPM algorithm. Compared with the other codebook construction process in [6], the proposed framework shows great advantages:

a. The proposed scheme can remarkably reduce the codebook construction time with only a tiny cost of annotation accuracy.
b. As the AP algorithm can automatically determine the quantization vector number, the size of codebook used in ScSPM can be self-adjusted by the algorithm base on the data. In this sense, the expansibility of the algorithm is enhanced.
c. With multi-level AP algorithm, the size of the codebook can be maximally reduced with acceptable drop in annotation accuracy. And the shrink of codebook size will shorten the SVM training time.

In this work, traditional AP algorithm is used to train the codebook. As it is known, the performance of AP algorithm is controlled by the input "Similarity" and "Preference" matrix as well as some other parameters. In the future, by adjusting those parameters, the performance of AP-ScSPM framework maybe enhanced.

References

1. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2169–2178 (2006)
2. Liang, M.Y., Du, J.P., Jia, Y.M., Sun, Z.Q.: Image Semantic Description and Automatic Semantic Annotation. In: International Conference on Control, Automation and Systems, pp. 1192–1195 (2010)
3. Perronnin, F.: Universal and Adapted Vocabularies for Generic Visual Categorization. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(7), 1243–1256 (2008)
4. Monay, F., Gatica-Perez, D.: Modeling Semantic Aspects for Cross-Media Image Indexing. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(10), 1802–1817 (2007)

5. Carneiro, G., Chan, A.B., Moreno, P.J., Vasconcelos, N.: Supervised Learning of Semantic Classes for Image Annotation and Retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(3), 394–410 (2007)
6. Yang, J.C., Yu, K., Gong, Y.H., Huang, T.: Linear Spatial Pyramid Matching Using Sparse Coding for Image Classification. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1794–1801 (2009)
7. Lin, S., Yao, Y., Guo, P.: Speed Up Image Annotation Based on LVQ Technique with Affinity Propagation Algorithm. In: Proceedings of the International Conference on Neural Information Processing: Models and Applications, pp. 533–540 (2010)
8. Frey, B.J., Dueck, D.: Clustering by Passing Messages between Data Points. Science 315(5814), 972–976 (2007)
9. Jiang, W., Ding, F., Xiang, Q.L.: An Affinity Propagation Based method for Vector Quantization Codebook Design. In: International Conference on Pattern Recognition, pp. 1–4 (2008)
10. Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., Poggio, T.: Robust Object Recognition with Cortex-Like Mechanisms. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(3), 411–426 (2007)
11. Mutch, J., Lowe, D.: Object Class Recognition and Localization Using Sparse Features with Limited Receptive Fields. International Journal of Computer Vision 80(1), 45–57 (2008)
12. Viitaniemi, V., Laaksonen, J.: Evaluating the performance in automatic image annotation: Example case by adaptive fusion of global image features. Image Communication 22(6), 557–568 (2007)
13. Jiang, Z., He, J., Guo, P.: Feature Data Optimization with LVQ Technique in Semantic Image Annotation. In: International Conference on Intelligent Systems Design and Applications, pp. 906–911 (2010)
14. Zhang, H., Berg, A., Maire, M., Malik, J.: SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2126–2136 (2006)
15. Everingham, M., Gool, L.V., Williams, C., Winn, J., Zisserman, A.: The pascal visual object classes challenge 2008. In: ECCV Workshop (2008)
16. Lee, H., Battle, A., Raina, R., Ng, A.Y.: Efficient sparse coding algorithms. In: Neural Information Processing Systems, pp. 801–808 (2006)
17. Mairal, J., Bach, F., Ponce, J., Sapiro, G., Zisserman, A.: Supervised dictionary learning. In: Neural Information Processing Systems (2009)

Airport Detection in Remote Sensing Images Based on Visual Attention

Xin Wang[1], Bin Wang[1,2], and Liming Zhang[1]

[1] Department of Electronic Engineering, Fudan University, Shanghai, 200433, China
[2] The Key Laboratory of Wave Scattering and Remote Sensing Information
(Ministry of Education), Fudan University, Shanghai, 200433, China
{09210720037,wangbin,lmzhang}@fudan.edu.cn

Abstract. This paper proposes an airport detection and recognition method for remote sensing image based on visual attention mechanism. Considering the disadvantage in traditional methods by which the remote sensing images are analyzed pixel by pixel, we introduce visual attention models into airport detection and improve the efficiency of automatic target detection greatly. Firstly, Hough transform is used to judge the existence of an airport and then the improved graph-based visual saliency (GBVS) visual attention model is used to extract regions of candidates (ROCs). According to the scale-invariant feature transform (SIFT) feature extracted from ROCs and classified by HDR tree, the airport areas are recognized. Experimental results show that the proposed method has faster speed, higher recognition rate and lower false alarm rate than other current methods, and is robust against white noise.

Keywords: Visual attention, Airport detection, SIFT feature, HDR tree, Hough transform.

1 Introduction

Recently, the automatic detection and recognition of targets in remote sensing images becomes more and more important, especially for airports, which are substantial objects in martial and civil area. However, the complex background of airports brings much difficulty to the detection. We summarize the keypoints in airport detection as follows: First, since airports locate in urban or suburban region, buildings, mountains, roads and some other natural or artificial objects always have similar features with airports, so we must find some unique and robust features to describe airports. Second, the basic feature of an airport is its runway, which consists of one or more straight lines, and has the stated length, width and intensity. At last, remote sensing images usually have large numbers of pixels, so we should pay attention to the complexity of methods in order to meet real-time processing request.

Previous works on airport detection can be mainly classified into two kinds [1]: edge detection-based and image segmentation-based. The former focused on the runway, using edge detection operation on an image and then finding straight lines by Hough transform [2] to locate the airport [1][3][4]. In [3], after the edge detection, the algorithm removed short or curled edges, then found long straight lines by Hough

transform and brought the region around the each line into SVM to recognize. However the latter concentrated on the textural difference between airports and surrounding, and applied image segmentation to extract ROCs [5]-[7]. The approach in [5] proposed in 2011 was a typical segmentation-based method and produced good results. It combined segmentation result and density of SIFT points to find ROCs, and then judged the location of the airport by statistical characteristics in ROCs such as mean, variance and moments.

There are both advantages and disadvantages in above methods. The edge detection-based method is simple and fast, but easily disturbed by the background. Besides, only using Hough transform to locate airports is not exact because the length of detected lines usually cannot match well with the one of runways. In contrast, the image segmentation-based way can locate the airport area suitably and reduce the influence of background. However, since the segmentation is executed pixel by pixel, this method is very slow and sensitive to parameters. Moreover, whether the selected statistical characteristics are robust to affine transformation is also a problem. In addition, it is noticeable that the false alarm rate of detection was not mentioned in most of previous works. In fact, it is important in practical applications that do not miss existing targets in images and do not recognize false targets.

Visual attention [8] is one of the most important parts of consciousness, and has been widely used in pattern recognition field and has received good results in processing of natural images. With the aid of visual attention, we can find salient objects quickly, which is different from machines. The visual attention mechanism consists of two processes: bottom-up and top-down. Bottom-up is the course which obtains a saliency map from an input image. It only relies on the input data. Nevertheless top-down has relation with one's knowledge and judgment, and it can modulate the saliency map generated by bottom-up attention. Because an airport has much difference with its circumambiency, we consider it to be salient. Therefore we introduce the visual attention mechanism into the detection of airports to extract ROCs. By this means, we overcome the disadvantage of low speed caused by pixel-to-pixel analysis and the problem of imprecise location of airports, and we can improve the efficiency of detection much without reducing detection accuracy.

Up to now, lots of visual attention models have been proposed such as NVT [9], STB [10] and AIM [11], but they are difficult to apply in real time due to the high complexity. Later, models in frequency domain, for example, SR [12], PFT [13] and PQFT [14], were proposed by Hou and Guo. However, these above models can receive preferable results on natural images but not for more complicated remote sensing images. In contrast, GBVS [15] is competent for target detection in complex background although with the shortcoming of low speed.

In this paper an improved GBVS model is suggested to speed up its computation and to be more suitable for airport detection. To be more specific, we use the improved GBVS algorithm to compute the saliency map of the input image and locate ROCs according to the map. Then we extract SIFT [16] feature on ROCs and take them into trained HDR [17] tree to classify. Finally we ensure the position of the airport among all the ROCs by the number of SIFT features which belong to airport in each ROC and the order of saliency.

2   Improved GBVS Model

The GBVS is derived from the NVT model. It constructs a Markov chain on the image and computes the equilibrium distribution of the chain to get the saliency map. It can be described as follows:

Step 1: Obtaining the Intensity Channel

Given an image I, use the Gaussian pyramid low-pass filters to process it. Each layer of the pyramid is a 2-D Gaussian low-pass filter G showed in (1):

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right), \tag{1}$$

where (x, y) represents the position of a pixel in I and σ is called the scale parameter. The larger σ is, the larger the range of smoothing. To build the pyramid, the original image is filtered and down-sampled again and again along with decrease of σ. As a result, a group of filtered results under different scales can be obtained and defined as the intensity channel.

Step 2: Obtaining the Orientation Channel

Like step 1, use the Gabor pyramid filters instead of the Gaussian to get the orientation information. The Gabor filter H is described as (2):

$$H(x, y, \sigma, \theta) = \frac{1}{\sigma^2} \exp\left(-\pi \frac{x^2 + y^2}{\sigma^2}\right) \left[\exp(i2\pi f(x\cos\theta + y\sin\theta)) - \exp(-\frac{\pi^2}{2}) \right], \tag{2}$$

here σ is the scale parameter, f is the frequency of the sine wave and θ is the orientation. Generally, the value of θ is

$$\theta = \left[0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\right]. \tag{3}$$

It means that the filters act in four directions. So four groups of results are produced and called the orientation channel.

Step 3: Computing the Feature Map

Denote any result map at all the scales of two channels as M. The dissimilarity d of two pixels $M(i, j)$ and $M(p, q)$ in a map can be defined as:

$$d((i, j) \| (p, q)) = \left|\log \frac{M(i, j)}{M(p, q)}\right|. \tag{4}$$

According to (4), the dissimilarity between any two pixels in M can be calculated. Then a fully-connected directed graph can be constructed on M. Each node of the graph is a pixel in M and each edge of it is the weight w between two pixels which is assigned by (5):

$$w((i, j), (p, q)) = d((i, j) \| (p, q)) \cdot \exp(-\frac{(i - p)^2 + (j - q)^2}{2\sigma^2}). \tag{5}$$

From (5) it is known that the weight w is determined by the distance and dissimilarity of pixels. Now a Markov chain is defined on the graph and the equilibrium distribution of it is calculated, which is the principal eigenvector with the largest eigenvalue of the weight matrix. The result is regarded as the feature map.

Step 4: Creating the Saliency Map

After getting all the feature maps in each channel, sum them together and normalize it. So there is a normalized map in intensity and orientation channel respectively. Eventually combine these two channels together as the saliency map.

The main problem of GBVS is speed which is mentioned in [13] and [14]. To solve this problem and aim at the airport detection, we improved the GBVS model as follows:

(1) Airports always have special size and there is not much difference in the size of different airports. So we use only two scales in GBVS, which is enough.

(2) An airport has better directivity thanks to its runway. To catch this feature, we increase the number of direction of Gabor filters from 4 to 8.

(3) We do the Hough transform on the original image and treat the result as an extra channel which will add into the model. We mark the detected lines with '1', others with '0', and filter it with Gaussian filters. The result is called the Hough channel and is summed and normalized with the other two channels.

(4) In order to exclude the interference from darker objects such as rivers and gorges, we multiply the saliency map with the original image I and set the result as the final saliency map. This operation can highlights an airport because airports in remote sensing images usually have higher brightness. Maybe this step belongs to the top-down process.

3   SIFT Feature

The selection of features used to describe a region is a key problem in pattern recognition. Familiar features are mainly based on statistics, such as mean, variance and moments. But some of them lack uniqueness, some are sensitive to affine transformation, and some others have high complexity which does not suit for rapid detection. SIFT was used to locate airports in [5] and received ideal results. So, we adopt this strategy in our method.

SIFT [7] was proposed by D. G. Lowe in 2004. It finds extremes in an image and creates feature vectors according to the information provided by the extreme and its surrounding. Compared with other features, SIFT is robust to affine transformation and fast. Details about SIFT can be found in [7]. In the course of SIFT extraction, the applications of different scales and rotation of the coordinate axis guarantee the invariance to affine transformation. We find that if we extract SIFT feature on airports, the keypoints often fall on the corner or junction of runways. By this token, the use of SIFT catches the basic feature of airports well.

4   HDR Tree

For target recognition, selection of classifier and training templates are indispensable. SVM is a general method which was used in [3] and [5] to recognize airports. However,

here we use HDR [17] tree instead. HDR tree is a kind of memory tree in which learning and search are very quick. Compared with SVM, HDR tree also possesses high speed in recognition of targets. And what is more, it can update its structure when new samples input without training all the samples again like SVM. The details of HDR tree was presented in [17]. When a new sample comes, it will find the leaf node which is most similar to it and be combined with the node. The node will split if necessary.

5   Airport Detection Based on Visual Attention

We use the improved GBVS model to extract ROCs and recognize airports with SIFT feature and HDR tree. This procedure can be shown as follows:

Step 1: Training

We select some images from our data set and extract SIFT features of them as training samples. We label the sample into two kinds: '1' means it belongs to airport and '0' points to non-airport class. Then we take the labeled samples into HDR tree for training.

Step 2: Preprocessing

For an input image, we should judge whether it contains an airport first in order to reduce unnecessary analysis. This is done by using Hough transform to the binary map of the image. If there is any detected line long enough, we will compute the saliency of the image; otherwise we will discard it directly. It is noticeable that Hough transform is used to make sure the existence of an airport but not to locate an airport as in [3]. Besides, the results of Hough transform will be directly brought in to the improved GBVS model.

Step 3: Creating a Saliency Map

Use the improved GBVS model to obtain the saliency map.

Step 4: Locating ROCs

We note the maximum value of the original saliency map is I_o and the maximum value of the current saliency map is I_c. At the beginning I_c is equal to I_o. Started from the brightest pixel in the saliency map, the region is growing to adjacency until the value of the edge pixel is smaller than $I_c \times \alpha$. α is a parameter valued in the range of [0, 1]. Then we draw the external rectangle of the grown region and treat the area in it as a ROC. After this, we set the pixel values in the ROC to zero and find the maximum value I_c of the current map. If I_c is below $I_o \times \alpha$, all the ROCs have been extracted and go to the next step; otherwise a next ROC is computed as before.

Step 5: Recognition

Extract SIFT features on each ROC and recognize them by HDR tree. Then each SIFT feature vector has a label '0' or '1'. Compute the feature rate of each ROC by (6):

$$\text{Feature rate} = \frac{\text{the number of label '1'}}{\text{the total number}} . \tag{6}$$

Step 6: Make Sure the Airport Region

Now each ROC has two attributes: the saliency order and the feature rate. We obtained the airport region in the following rule: If there is a ROC and the corresponding feature rate of which is higher than a threshold β, it will be considered as the airport region; otherwise we pay attention to the ROCs one by one according to the saliency order. If there is one or more feature vectors labeled '1', it will be defined as the airport area without considering the latter ROCs.

The flow chart of our method is showed in Fig. 1.
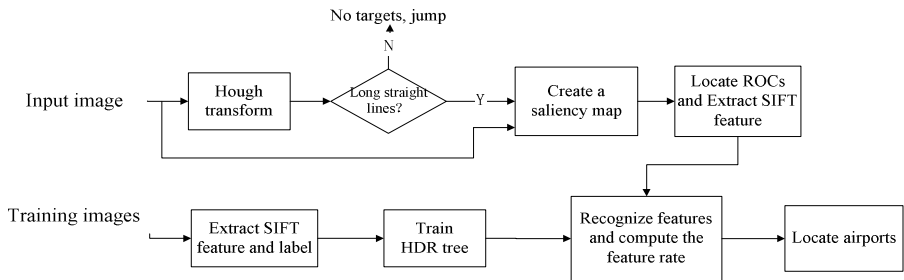
Fig. 1. The flow chart of the algorithm

6 Experimental Results

We demonstrate the performance of our method using a set of data which comes from Google Earth. The data in Google Earth are combination of aerial photographs and satellite images which are mainly collected by Landsat-7. For our experiments, we choose 200 color images in 30m×30m resolution with the size of 400×400, and half of them contains an airport. We change all of the images into gray and take 7 of them as training set while others for test. The environment of our experiments is MatlabR2008a in such computer as Inter Core2 2.53GHz CPU and 2G Memory.

Our method of airport detection can be divided into four main phases: original map, saliency map, ROCs and the final recognized area. An example of this process is shown in Fig.2. We choose the parameters as $\alpha = 0.35$, $\beta = 0.2$.

From Fig. 2 we can find that although rivers are selected as ROCs together with the airport, they can still be distinguished from the airport well through SIFT feature. Some results of airport detection are presented in Fig. 3. Fig. 3 illustrates that our method can locate airports among many kinds of interferers such as mountains, clouds and coast.

(a)

(b)

(c)

(d)
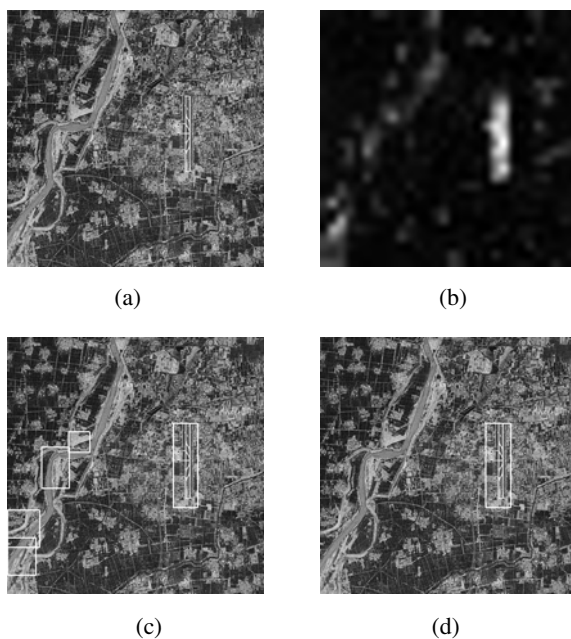
Fig. 2. Results step by step: Jinan, China (a) Original map; (b) saliency map; (c) ROCs; (d) airport region

(a)

(b)

(c)

(d)

(e)

(f)
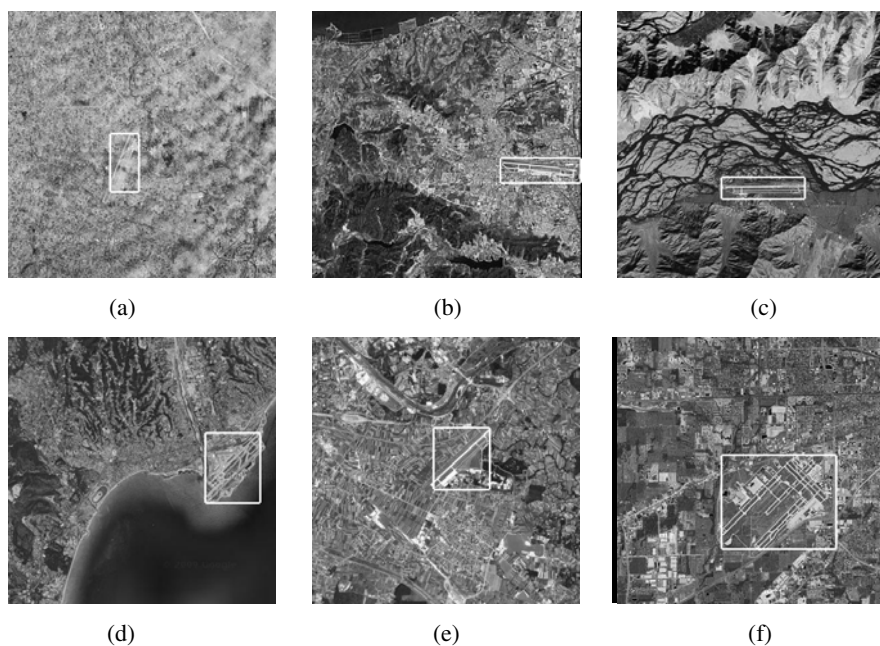
Fig. 3. Some examples: (a) Chengdu, China; (b) Dalian, China; (c) Lasa, China; (d) Nisi, France; (e) Zagreb, Croatia; (f) Indiana, USA

In addition, we show some failure detection cases in Fig. 4. It can be obtained that long and straight roads which have similar features with airports will disturb our detection in spite of the training of airports.

(a) Houston, USA   (b) Beihai, China
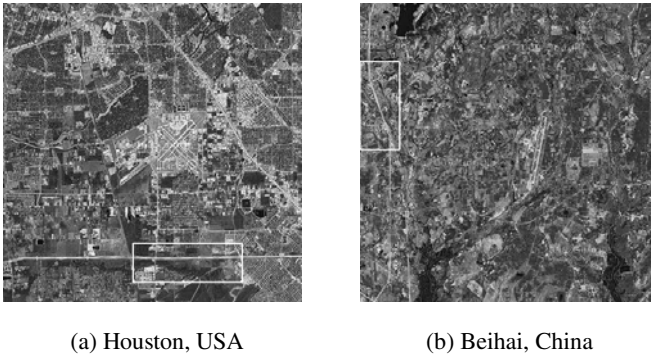
Fig. 4. Two failure examples: (a) Houston, USA; (b) Beihai, China

We compared with the approaches proposed in [3] and [5]. The parameters in [5] are selected as R=80 and S=3. R means the radius of the SIFT cluster and S is the minimum number of keypoints in one cluster. Our data set is tested by the above three methods and 30 images are chosen as training set for approaches in [3] and [5] to obtain best results. We analyzed the performance in three aspects: the recognition rate, the false alarm rate and time consumption. The results are shown in Table 1.

Table 1. Comparison between three methods

Methods	The recognition rate	The false alarm rate	Time
Literature [5]	66%	28%	More than 10min
Literature [3]	65%	18%	2.43s
Ours	**91%**	**10%**	**2.59s**

Table 1 show that our method has obvious advantages relative to the others. The problems of [5] and our difference from it can be summarized as follows: First, it is difficult to find a fixed value of parameters R and S which are suitable for different size of airports, but our method is robust to the size change because it is based on the scale-space. Second, the segmentation method which is adopted to extract ROCs in [5] consumes a mass of time and by contraries the improved GBVS model is computed quickly. At last, the texture feature used for recognition is not robust to affine transformation, especially for the Zernike [18] moment. The application of higher order of Zernike moment will solve this problem but with the shortcoming of low speed. However we use SIFT feature to recognize airports without consideration of this. In addition, the main trouble of [3] is that it is difficult to detect inclined airports which are not vertical and horizontal to the coordinate axes, but our Gabor filters avoid this problem.

Receiver operating characteristic (ROC) curve is recently used to evaluate the efficiency of detection. The x-axis of the curve is the false positive rate (FPR) and

the y-axis is the true positive rate (TPR). The larger the ROC area is, the better the efficiency is. So we plot ROC curves of the above three methods in Fig.5 (a). As can be seen, our method has largest ROC area comparing to [5] and [3].
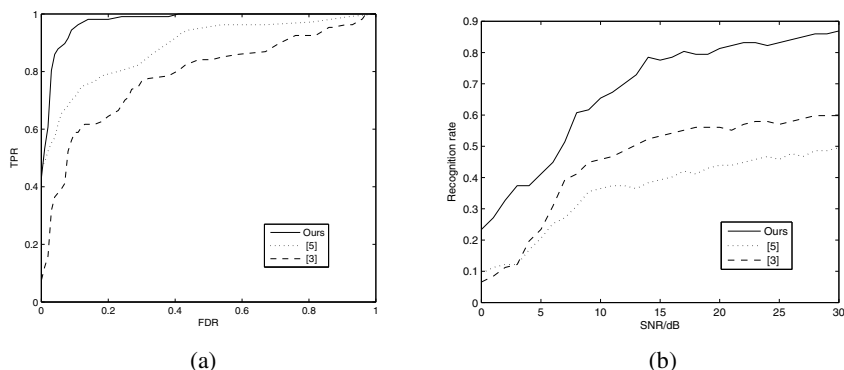
Fig. 5. (a) The ROC curve of the methods; (b) The recognition rate under noise

Additive noise is inevitable in remote sensing images, so we added white Gaussian noise on test images and discuss the performance under noisy data. The SNR is increased from 0dB to 30dB, and the curve of recognition rate versus the SNR is plot in Fig. 5 (b). The recognition rate of our method becomes steady after 15dB and higher than the others. Besides, the algorithm in [3] is less influenced than the one in [5] because when the power of noise is not too large, the influence of noise is limited for the detection of long straight lines which is used to locate airports.

7   Conclusions

This paper introduced visual attention mechanism into detection of airport target in remote sensing images, and proposed a new method to detect airports from remote sensing images which is based on computational model of visual attention. Through this method, the shortcoming caused by pixel-to-pixel analysis in traditional methods is overcome and the efficiency of detection is improved a lot. Experimental results show that our method has faster speed, higher recognition rate and lower false alarm rate than the previous works. It should be very useful in real-time target detection, especially when the background is complicated.

References

1. Wang, W., Li, L., Hu, C.B., et al.: Airport Detection in SAR Image Based on Perceptual Organization. In: International Workshop on Multi-Platform/Multi-sensor Remote Sensing and Mapping (M2RSM), pp. 1–5 (2011)
2. Duda, R.O., Hart, P.E.: Use of the Hough Transformation to Detect Lines and Curves in Pictures. CACM 15, 11–15 (1972)

3. Qu, Y.Y., Li, C.H., Zheng, N.N.: Airport Detection Base on Support Vector Machine from a Single Image. In: Proc. ICICS, pp. 546–549 (2005)
4. Pi, Y.M., Fan, L.H., Yang, X.B.: Airport Detection and Runway Recognition in SAR Images. In: Proc. IEEE Int. Geosci. Remote Sens. Symp., pp. 4007–4009 (2003)
5. Tao, C., Tan, Y.H., Cai, H.J., Tian, J.W.: Airport Detection from Large IKONOS Images Using Clustered SIFT Keypoints and Region Information. IEEE Geosci. Remote Sens. Lett. 8, 128–132 (2011)
6. Liu, D.H., He, L.B., Carin, L.: Airport Detection in Large Aerial Optical Imagery. In: IEEE International Conference on Acoutics, Speech and Signal Processing, vol. 5, pp. 761–764 (2004)
7. Zhou, N., Chen, W., et al.: Airport Detection Algorithm in Large Area Satellite Borne SAR Images. Systems Engineering and Electronics 31, 761–764 (2009)
8. Desimone, R., Duncan, J.: Neural Mechanisms of Selective Visual Attention. Annual Reviews in Neuroscience 18, 193–222 (1995)
9. Itti, L., Koch, C., Niebur, E., et al.: A Model of Saliency-Based Visual Attention for Rapid Scene Analysis. IEEE Trans. Patt. Anal. and Mach. Intell. 20, 1254–1259 (1998)
10. Walther, D., Koch, C.: Modeling Attention to Salient Proto-Objects. Neural Networks 19, 1395–1407 (2006)
11. Bruce, N.D., Tsotsos, J.K.: Saliency Based on Information Maximization. In: NIPS (2005)
12. Hou, X.D., Zhang, L.Q.: Saliency Detection: A Spectral Residual Approach. In: CVPR (2007)
13. Guo, C.L., Ma, Q., Zhang, L.M.: Spatio-Temporal Saliency Detection Using Phase Spectrum of Quaternion Fourier Transform. In: CVPR (2008)
14. Guo, C.L., Zhang, L.M.: A Novel Multi-resolution Spatiotemporal Saliency Detection Model and Its Applications in Image and Video Compression. IEEE Trans. Image Proc. 19, 185–198 (2010)
15. Harel, J., Koch, C., Perona, P.: Graph-Based Visual Saliency. In: NIPS (2006)
16. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60, 91–110 (2004)
17. Hwang, W.S., Weng, J.Y.: Hierarchical Discriminate Regression. IEEE Trans. Patt. Anal. and Mach. Intell. 22, 1277–1293 (2000)
18. Khotanzad, A., Hong, Y.H.: Invariant Image Recognition by Zernike Moments. IEEE Trans. Patt. Anal. and Mach. Intell. 12, 489–497 (1990)

A Method to Construct Visual Recognition Algorithms on the Basis of Neural Activity Data

Hiroki Kurashige and Hideyuki Câteau

BSI-TOYOTA Collaboration Center, Brain Science Institute, RIKEN
Hirosawa 2-1, Wako-shi, 351-0198 Japan
{kura,cateau}@brain.riken.jp

Abstract. Visual recognition by animals significantly outperforms man-made algorithms. The brain's intelligent choice of visual features is considered to be underlying this performance gap. In order to attain better performance for man-made algorithms, we suggest using the visual features that are used in the brain in these algorithms. For this goal, we propose to obtain visual features correlated with the brain activity by applying a kernel canonical correlation analysis (KCCA) method to pairs of image data and neural data recorded from the brain of an animal exposed to the images. It is expected that only the visual features that are highly correlated with the neural activity provide useful information for visual recognition. Applied to hand-written digits as image data and activity data of a multi-layer neural network model as a model for a brain, the method successfully extracted visual features used in the neural network model. Indeed, the use of these visual features in the support vector machine (SVM) made it possible to discriminate the hand-written digits. Since this discrimination required to utilize the knowledge possessed in the neural network model, a simple application of the usual SVM without the use of these features could not discriminate them. We further demonstrate that even the use of non-digit hand-written characters for the KCCA extracts visual features which enable the SVM to discriminate the hand-written digits. This indicates the versatile applicability of our method.

Keywords: Brain-based visual recognition algorithm, Kernel canonical correlation analysis, Multi-neuron activity, Visual features.

1 Introduction

The visual recognition of specific and generic objects by machines would have wide applications for car-mounted pedestrian detectors, surveillance systems, and vision-based robotics, etc. However, in order to design a high-performance visual recognition algorithm, a proper choice of visual features is essential. We assume that the animal's brain uses the useful visual features for their visual recognition so that they can fulfill complex visual recognition effortlessly. Hence, we suggest using such visual features in the visual recognition algorithm. Recent advances in neuroscience have allowed the recording of neural activity data from

large neural populations of a brain of a living animal with high spatio-temporal resolution [1,2,3]. To extract the visual features that the brain of the animal uses from the neural activity data taken with these recording techiniques, we propose a concrete method to achieve this for given brain activity data. The extracted features are then incorporated into visual recognition algorithms.

To evaluate the potential of our method, we applied our method to activity data recorded from a model brain, an artificial neural network (ANN) that discriminates illegible hand-written digits. Our method successfully extracted the visual features used in the ANN from the simulated neural activity data. Indeed, a visual recognition algorithm based on the support vector machine (SVM) that incorporated the extracted features showed an almost perfect performance in a task which required the use of the visual features that the ANN used. In contrast, the usual SVM performed at a nearly chance-level success rate. The neural activity data recorded from an ANN, although they are artificial, share several important characters with the real neural data. Especially, both our ANN and the real brain represent each visual information as the distributed population activity, not as the single neuronal activity (see [4]).

In our method, we utilize neural activity data aiming at best possible extraction of the useful visual features used in an animal's brain. There are several factors worth noting to achieve this goal. First, since experimental observations suggest that various visual features are represented by neural population activity [4], we need to extract the visual features from population activity data. Next, we need to read off the nonlinear relationship between images and neural activity because of the nonlinear correspondence between images and neural activity representing the images [5]. Moreover, we should express the visual features with a nonparametric formularion because it is difficult to express high-level visual features characterizing objects or object categories. In contrast, low-level physical visual features, e.g. orientation and scale of edge, are well expressed by simple functions like the Gabor function [6]. Here, we adopt a kernel canonical correlation analysis (KCCA) method [7,8] to extract useful visual features from paired data of images and neural activities. KCCA fits the requirements noted above. KCCA has wide applications in various fields [9,10] including neuroscience [11,12,13]. Finally, these extracted visual features are used in kernel-based visual recognition algorithms like SVM. In contrast to our method, existing methods [14,15,16] that use neuroscientific knowledge to develop visual recognition algorithms extract only the low-level features from neural data or try to mimic the qualitative aspects of the brain such as a rough architecture of the information processing.

In section 2, we first overview our method, explain KCCA, and provide a special kernel playing a main role to provide the visual features that a brain uses to visual recognition algorithm. In section 3, we illustrate a framework to evaluate our method, discuss the problem that we call *ignorance of variation* encountered in real-world applications of machine learning, and show how our method can overcome this problem. Section 4 is devoted to summarize the result, implications, and future directions of our method.
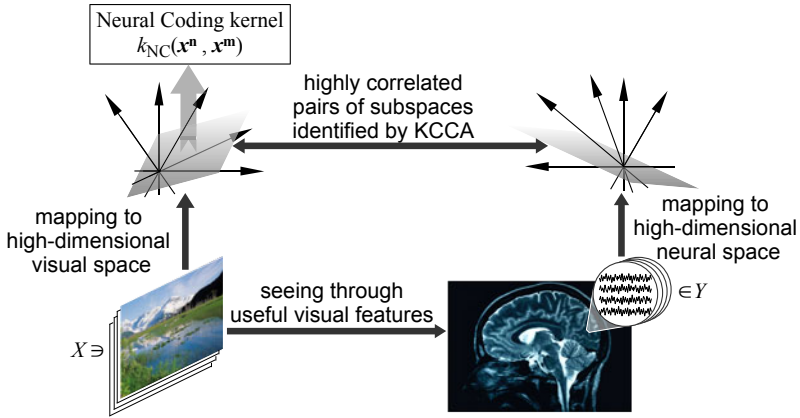
Fig. 1. Proposed Method. Apply KCCA to correlate images with neural activities recorded from a brain of an animal exposed to the images for construction of an NC kernel.

2   Methods

Visual features that correlate well with the neural activity are assumed to be the ones that the brain uses, while visual features that are poorly correlated with neural activity are what the brain is likely to neglect. Our strategy to identify useful visual features is therefore to correlate images (Fig. 1, lower left) with the neural activity elicited when an animal is exposed to the images (Fig. 1, lower right). To a given set of N images, $X = \{\mathbf{x^1}, \cdots, \mathbf{x^N}\}$, and a corresponding set of the neural activities, $Y = \{\mathbf{y^1}, \cdots, \mathbf{y^N}\}$, both in vector representations, we apply a kernelized version of canonical correlation analysis (CCA) to identify visual features that correlate well with neural activity. CCA is a linear multivariate statistical method to identify pairs of vectors, $\{(\mathbf{u^{(j)}}, \mathbf{v^{(j)}})\}_j$ such that the projections of $\{\mathbf{x^1}, \cdots, \mathbf{x^N}\}$ onto $\mathbf{u^{(j)}}$ defined with $f_{\mathbf{u^{(j)}}}(\mathbf{x}) = \mathbf{u^{(j)}}^{\mathrm{T}}\mathbf{x}$ are highly correlated with the projections of $\{\mathbf{y^1}, \cdots, \mathbf{y^N}\}$ onto $\mathbf{v^{(j)}}$ defined with $g_{\mathbf{v^{(j)}}}(\mathbf{y}) = \mathbf{v^{(j)}}^{\mathrm{T}}\mathbf{y}$.

A kernelized version of CCA, called KCCA [7,8], identifies such correlated pairs of projections in the high dimensional visual and neural spaces (Fig. 1, mid left and mid right, respectively), so that it can account for possible nonlinear correlations between the pair of the original data. The left and right gray planes in Fig. 1 represents the subspaces spanned respectively by the identified vecrors $\{\mathbf{u^{(j)}}\}_{j=1,2,\cdots}$ and $\{\mathbf{v^{(j)}}\}_{j=1,2,\cdots}$ in high dimensional spaces. Those set of vectors respectively represent useful visual features and the neural representation of the visual features in the high dimensional spaces.

Considering the theory of reproducing kernel Hilbert space [8], the projection in the high dimensional visual space is rewritten as $f_{\boldsymbol{\alpha}}(\mathbf{x}) = \sum_{n=1}^{N} \alpha_n k_X(\mathbf{x}, \mathbf{x^n})$ with kernel funciton, $k_X(\mathbf{x}, \mathbf{x'})$. Similarly, we have $g_{\boldsymbol{\beta}}(\mathbf{y}) = \sum_{n=1}^{N} \beta_n k_Y(\mathbf{y}, \mathbf{y^n})$,

for the projection in the high dimensional neural space. We used the standard Gaussian kernel for $k_X(\mathbf{x}, \mathbf{x^n})$ and $k_Y(\mathbf{y}, \mathbf{y^n})$ because of its universal capability in function approximation [17]. Therefore, our task in this formulation is to find $(\boldsymbol{\alpha}, \boldsymbol{\beta})$ instead of $(\mathbf{u}, \mathbf{v})$. To find $(\boldsymbol{\alpha}, \boldsymbol{\beta})$, we maximize the regularized correlation:

$$\max_{\boldsymbol{\alpha}, \boldsymbol{\beta}} \frac{\boldsymbol{\alpha}^{\mathrm{T}} K_X K_Y \boldsymbol{\beta}}{\sqrt{\boldsymbol{\alpha}^{\mathrm{T}}(K_X^2 + \zeta_x K_X)\boldsymbol{\alpha}}\sqrt{\boldsymbol{\beta}^{\mathrm{T}}(K_Y^2 + \zeta_y K_Y)\boldsymbol{\beta}}}$$

with K_X and K_Y being the Gram matrices for $k_X(\mathbf{x}, \mathbf{x'})$ and $k_Y(\mathbf{y}, \mathbf{y'})$, and ζ_x and ζ_y being regularization parameters to avoid overfitting. The maximization problem is known to be equivalent to the generalized eigenvalue problem:
$$\begin{pmatrix} 0 & K_X K_Y \\ K_Y K_X & 0 \end{pmatrix}\begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{pmatrix} = \lambda \begin{pmatrix} K_X^2 + \zeta_x K_X & 0 \\ 0 & K_Y^2 + \zeta_y K_Y \end{pmatrix}\begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{pmatrix}.$$ A jth eigenvector $(\boldsymbol{\alpha^{(j)}}^T, \boldsymbol{\beta^{(j)}}^T)^T$ defines the jth pair of visual and neural canonical variables, $f_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}) = \sum_{n=1}^{N} \alpha_n^{(j)} k_X(\mathbf{x}, \mathbf{x^n})$ and $g_{\boldsymbol{\beta}^{(j)}}(\mathbf{y}) = \sum_{n=1}^{N} \beta_n^{(j)} k_Y(\mathbf{y}, \mathbf{y^n})$. All the visual (neural) canonical variables are constructed to be mutually uncorrelated. We selected only pairs of canonical variables of which the correlation value is greater than 0.7 to obtain highly correlated pairs of the visual and neural canonical variables. The correlation value was calculated with the data set without the data used to solve the above generalized eigenvalue problem to avoid obtaining wrongly high values due to the overfitting. The resultant d pairs of visual and neural canonical variables, $\{(f_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x}), g_{\boldsymbol{\beta}^{(j)}}(\mathbf{y}))\}_{j=1,\cdots,d}$ represent highly correlated pairs of the visual features and the neural representations of them.

Our idea is that the visual features that correlate well with the neural activity are used in the brain. Such visual features must be useful for visual recognition. Therefore, we call such visual canonical variables $\{(f_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x})\}_{j=1,\cdots,d}$ the *useful visual features*. Our proposal is to perform visual recognition with the SVM or other kernel-based algorithms not within the entire feature space but in the restricted space spanned by useful visual features. We naturally expect that the visual discrimination in this space performs better.

Neural Coding Kernel
The restriction to the useful visual features implies the use of a special kernel function that represents the similarity between images $\mathbf{x}$ and $\mathbf{x'}$ by the similarity in the space spanned by d useful visual features, $\{f_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x})\}_{j=1,\cdots,d}$. We call this kernel *Neural Coding* (NC) kernel or k_{NC} (Fig. 1). Here, $q(\mathbf{x}) = (f_{\boldsymbol{\alpha}^{(1)}}(\mathbf{x}), \cdots, f_{\boldsymbol{\alpha}^{(d)}}(\mathbf{x}))^{\mathrm{T}}$ represents the mapping of images to the space spanned by the useful visual features, where we expect that visual discrimination is efficient. Specifically, the NC kernel is defined as,

$$k_{\mathrm{NC}}(\mathbf{x}, \mathbf{x'}) \equiv \exp(-\frac{|q(\mathbf{x}) - q(\mathbf{x'})|^2}{2\sigma^2}). \tag{1}$$

An NC kernel accounts for the brain-based similarity of images so that this kernel function makes kernel-based algorithms such as SVM closer to the brain. Although we used the Gaussian kernel, $\exp(-\frac{|\bullet|^2}{2\sigma^2})$, to represent the similarity
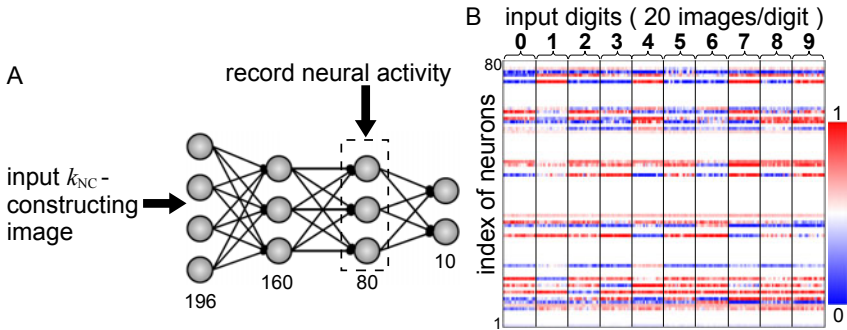
Fig. 2. Neural activity recorded from ANN. (A) Feedforward architecture of the ANN illustrating where a k_{NC}-constructing image comes in and which neural activity is recorded. Each number represents the number of neurons in each layer. (B) Neural activity recorded from the third layer of the ANN. Each row represents in color code the activity of each neuron in the third layer for presented normal and distorted digits, 20 different images per digit. Digits described at upper-side are probided to the ANN.

between $q(\mathbf{x})$ and $q(\mathbf{x}')$ (see (1)) in the present study, arbitrary positive semidefinite kernels (e.g. linear kernel, polynomial kernel, etc) can be used.

3   Evaluations of the NC-Kernel Method

Visual Recognition Task

Here, we used a model brain to test how our method of extracting visual features used in the brain worked. As a model brain, we used a four-layer (196-160-80-10 neurons) ANN model (Fig. 2A) with regularization [18]. Our ANN had been trained well to perform difficult multi-class discrimination of hand-written digits ('0' to '9') at a high success rate. The ten neurons in the ANN's output (fourth) layer were intended to address which digit was provided to the input layer. For example, when a hand-written digit representing '5' was shown to the ANN, the fifth but not other output neurons was activated. The hand-written digits consisted of normal (Fig. 3, left upper) and distorted digits (Fig. 3, left lower). We confirmed that the ANN correctly discriminated digits in both types (Table 1, 'ANN'). Hand-written digits were 196-pixel images which had been downscaled from the original digit images stored in the ETL6 database of the Electrotechnical Laboratory in Japan (http://www.is.aist.go.jp/etlcdb/).

We first asked whether or not the usual SVM based on the standard Gaussian kernel showed good performance for discriminating a target digit (e.g. '5') from non-target digits (e.g. digits except '5'). Normal digits were used for training (100 positive and 100 negative samples), while distorted digits were used for the test (60 positive and 60 negative samples) (Table 1, 'SVM'). The very low pixel-based similarity between normal and distorted digits (see Fig. 3, left) made the usual SVM with the Gaussian kernel performed only at a near-chance

normal (upper) and distorted (lower) hand-written digits   hand-written hiragana
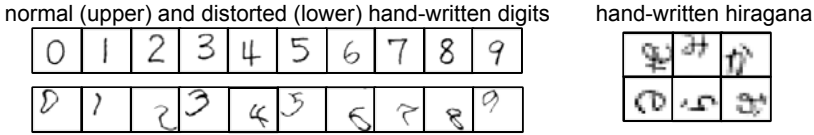
Fig. 3. Examples of hand-written characters. Digits (left, shown with original resolution) and *hiragana* (right, shown with downscaled resolution).

Table 1. Hand-written character image sets used in each procedure

	ANN training	ANN test	KCCA[*1]		SVM[*2] training	SVM[*2] test
			digits	hiragana		
normal digits	✓	✓	✓		✓	
distorted digits	✓	✓	✓			✓
hiragana				✓		

*1 These left and right columns correspond to Fig. 4A center and right, respectively.
*2 SVM here represents either the one based on the Gaussian kernel or the NC kernel.

level (Fig. 4A). This result suggests that additional information is required to successfully perform this discrimination. Therefore, we aimed to extract useful visual features as the additional information from the ANN using KCCA.

To extract visual features which were used in the ANN and to construct a NC kernel out of them, we first needed to prepare images to be shown to the ANN for us to record its neural activity. We call such images k_{NC}-*constructing images*, for which we used hand-written digits or Japanese characters, *hiragana*, in this study (Fig. 3). We recorded the neural activity at the third layer of ANN (Fig. 2A). We constructed a NC kernel by applying KCCA to pairs of images and the corresponding neural activities recorded from the ANN that received the image. Fig. 2B shows the neural activity we observed when the digits were input to the ANN as k_{NC}-constructing images. The observed neural activity was a distributed representation among the neural population, similar to what has been observed in the real brain [4].

Evaluation of our Method

In our evaluation of our method, we asked if useful visual features were efficiently extracted from the ANN so that they improved the performance of the SVM. We first used both normal and distorted hand-written digits as k_{NC}-constructing images (Table 1, left of 'KCCA'). As a result, the KCCA, through linearization in the high dimensional spaces, successfully identified high correlation between the visual and neural canonical variables, $f_{\boldsymbol{\alpha}^{(j)}}(\mathbf{x})$ and $g_{\boldsymbol{\beta}^{(j)}}(\mathbf{y})$. These obtained linear correlation (Fig. 4B, left) was originally nonlinear between the lower two spaces in Fig. 1. The extracted visual features were used to define the NC kernel as in
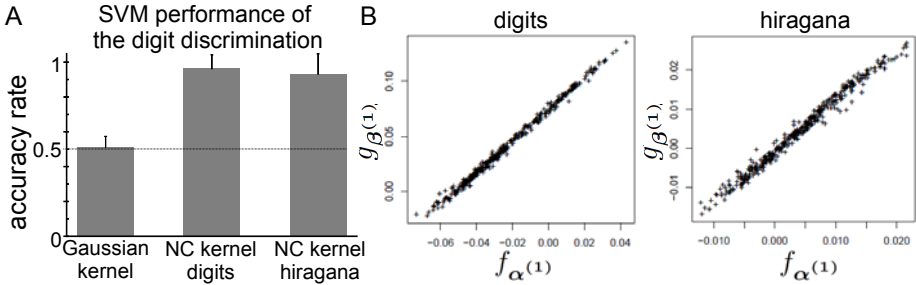
Fig. 4. Evaluation of our method. (A) Accuracy rate for solution of a discrimination task by SVM. (B) High correlation between visual and neural canonical variables, meaning successful identification of the relationship between images and neural activities.

([1](#)). The SVM based on the NC kernel achieved a success rate of 96% in comparison with the poor success rate of 51% achieved by the usual SVM (Fig. 4A). This evaluation with the ANN shows that our framework works well.

This high performance of the NC kernel-based SVM is understandable because the NC kernel was constructed by using the digit image set which covered the full variation of distorted digits used at a test (Table 1). However, we usually do not know in advance the range of variations in images which the SVM is supposed to handle at a test in future. We call this problem the *ignorance of variation*: an issue that frequently exists in real-world applications of machine learning. Thus we next asked if effective NC kernel could be constructed by using non-digit images as k_{NC}-constructing images.

Specifically, we asked if the use of hand-written Japanese characters, *hiragana* (Fig. 3, right), as k_{NC}-constructing images (Table 1, right of 'KCCA') would extract visual features that are useful for the digit discrimination. The hand-written *hiragana* characters were taken from the ETL7L database of the Electrotechnical Laboratory in Japan and were downscaled to 196-pixel. We applied the various transformation (reflections across vertical, horizontal and/or diagonal axes) to the *hiragana* characters to increase their variation. However, in order to make the *hiragana* characters really different from the distorted digits, we never applied the same distortion to hiragana as what we applied to the digits to obtain the distorted digits in Fig. 3 (lower). Our purpose here is to see if non-digit characters are usable in extracting useful visual features. Therefore, we needed to discard a hiragana characters that accidentally resembled to one of the digits. We thereby discarded a *hiragana* character if the activity elicited at the final fourth layer, $(n_0, n_1, \cdots, n_9)$, showed one-component dominance, namely only one of n_js is greater than 0.8 while all the other n_js are less than 0.2. Even with such hiragana characters that were carefully selected to be distant enough from digits, the NC kernel-based SVM achieved about 93% of an accuracy rate (Fig. 4A, see Fig. 4B, right). We note that *hiragana* characters were neither used in the ANN training/test nor SVM training/test (Table 1). These simulation

results suggest that images not specific to the prospective visual discrimination task, such as *hiragana* characters, can be used as k_{NC}-constructing images.

4 Discussion

In this study, we proposed a method to extract useful visual features using KCCA from the neural activity data recorded along with corresponding image data, and to represent the set of useful visual features as an NC kernel that can be used in the kernel methods. We showed that the visual features useful to the visual recognition task were actually transferred from an ANN to the SVM via the NC kernel. Importantly, to improve the SVM discrimination of some data set with this method, the k_{NC}-constructing images should not necessarily coincide with the data set to be discriminated. In fact, we demonstrated that k_{NC} constructed from a set of *hiragana* characters dramatically improved the discrimination performance of digits.

The visual recognition by an animal is much more flexible and accurate than the one by the ANN used in this study. However, the ANN and the brain of the animal share the properties that are related to the success and failure of the extraction of useful visual features: nonlinearity in information processing [5] and distributed nature of the information representation [19]. Our results demonstrated that KCCA is a suitable tool to extract information from such nonlinear and distributed activity data since the KCCA is designed to extract "many-to-many" nonlinear correlations.

As other nonlinear kernel-based methods, explicit representations of the extracted features are difficult in our method (pre-image problem). However, our method is meant to provide an improved visual discrimination algorithm without knowing the explicit representations of the features.

We showed that a set of hiragana characters worked well as k_{NC}-constructing images, which we consider was because *hiragana* characters partially share visual features with digits. When we apply the present method to the data from real neuroscientific experiments using animals, what type of images can we use as k_{NC}-constructing images? We expect that good candidates are the so-called natural images [20] for the following reason. First of all, we note that images whose recognition can be improved by the present method are by definition of the method restricted to the objects that animals can recognize. Very strange images that never resemble to any existing object in nature and cannot be guaranteed to be recognizable by animals are out of the scope of the present method. What then are the characteristics of an object recognizable by animals? We consider that such a recognizable object, even if it is man-made, should share visual features with natural images such as scenes of forest, mountains, crops etc. It is because animals generalize their evolutionary and developmentally acquired ability to recognize natural images to the recognition of the object in question. Because of such shared features supposedly included in the natural images, we expect that the use of the natural images as k_{NC}-constructing images improves the performances of the object recognition algorithms, just as the use of *hiragana* characters improved the performance of the digit discrimination algorithm.

In contrast to our idea of using KCCA to extract nonlinear correlation between images and multi-neuron activity, Nishimoto et al. used a nonlinear fitting method to study the correlation between images and single-neuron activity [21]. Instead of designing a novel visual recognition algorithm, Fujiwara et al. aimed to construct a good brain machine interface to predict what human saw from his/her fMRI signals, and used a Bayesian CCA to find a good basis for visual image reconstruction [22].

Our method is versatile in two ways. First, the NC kernel is applicable to arbitrary kernel methods. Second, once a NC kernel is identified from natural images, it can be used to process a variety of images as long as their features are shared with natural images.

Although so far we tested our method on simulated data, we will test our method on neural data taken in our own laboratory or other laboratories in near future. For our method, the neural activity is supposed to be recorded from the brain of a living animal or human subject exposed to the natural images. Among several possible recording techniques, functional MRI [1] by which we record the neural activity from whole brain of an animal or human subject safely is promising. Another possibility is voltage-sensitive dye imaging [2] by which the neural activity is recorded from wide range of the brain at high temporal resolution with reasonable spatial resolution, although this technique cannot be applied to human subjects. In vivo two-photon Ca^{2+} imaging [3] is also a good possibility, which enable us to record neural activity at cellular-level spatial resolution although the spatial range recorded simultaneously is limited.

Additionally, there are several options for relevant recording areas of the brain. Generally, neurons sitting earlier in the brain visual pathway process localized and physical (e.g. orientation and spatial frequency) features, while neurons sitting later in the visual pathway process more global and abstract (e.g. object category) features. Therefore, we expect that resultant NC kernels will vary according to the difference in recording techniques and areas. In order to construct the visual recognition algorithm with the broadest utility, we may think of synthesizing an optimized kernel from multiple NC kernels constructed by different sets of recording data. This approach uses the framework of multiple kernel learning [23].

References

1. Logothetis, N.K.: What We Can Do and What We Cannot Do with fMRI. Nature 453, 869–878 (2008)
2. Ayzenshtat, I., Meirovithz, E., Edelman, H., Werner-Reiss, U., Bienenstock, E., Abeles, M., Slovin, H.: Precise Spatiotemporal Patterns among Visual Cortical Areas and Their Relation to Visual Stimulus Processing. J. Neurosci. 30, 11232–11245 (2010)
3. Cheng, A., Goncalves, J.T., Golshani, P., Arisaka, K., Portera-Cailliau, C.: Simultaneous Two-photon Calcium Imaging at Different Depths with Spatiotemporal Multiplexing. Nature Methods 8, 139–142 (2011)

4. Kiani, R., Esteky, H., Mirpour, K., Tanaka, K.: Object Category Structure in Response Patterns of Neuronal Population in Monkey Inferior Temporal Cortex. J. Neurophysiol. 97, 4296–4309 (2007)
5. Wu, M.C., David, S.V., Gallant, J.L.: Complete Functional Characterization of Sensory Neurons by System Identification. Annu. Rev. Neurosci. 29, 477–505 (2006)
6. Jones, J.P., Palmer, L.A.: An Evaluation of the Two-dimensional Gabor Filter Model of Simple Receptive Fields in Cat Striate Cortex. J. Neurophysiol. 58, 1233–1258 (1987)
7. Akaho, S.: A Kernel Method for Canonical Correlation Analysis. In: Proc. of International Meeting of the Psychometric Society (IMPS 2001). Springer, Heidelberg (2001)
8. Bach, F.R., Jordan, M.I.: Kernel Independent Component Analysis. J. Mach. Learn. Res. 3, 1–48 (2002)
9. Vinokourov, A., Shawe-Taylor, J., Cristianini, N.: Inferring a Semantic Representation of Text via Cross-language Correlation Analysis. In: Advances in Neural Information Processing Systems, vol. 15, pp. 1473–1480. MIT Press, Cambridge (2002)
10. Suetani, H., Iba, Y., Aihara, K.: Detecting Generalized Synchronization between Chaotic Signals: a Kernel-based Approach. J. Physics A 39, 10723–10742 (2006)
11. Hardoon, D.R., Mourao-Miranda, J., Brammer, M., Shawe-Taylor, J.: Unsupervised Analysis of fMRI Data using Kernel Canonical Correlation. Neuroimage 37, 1250–1259 (2007)
12. Macke, J.H., Zeck, G., Bethge, M.: Receptive Fields without Spike-triggering. In: Advances in Neural Information Processing Systems, vol. 21, pp. 969–976. MIT Press, Cambridge (2008)
13. Bießmann, F., Meinecke, F.C., Gretton, A., Rauch, A., Rainer, G., Logothetis, N.K., Müller, K.-R.: Temporal Kernel CCA and its Application in Multimodal Neuronal Data Analysis. Machine Learning 79, 5–27 (2010)
14. Fukushima, K.: Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position. Biol. Cybern. 36, 193–202 (1980)
15. LeCun, Y., Huang, F.J., Bottou, L.: Learning Methods for Generic Object Recognition with Invariance to Pose and Lighting. In: CVPR, pp. 97–104. IEEE Press (2004)
16. Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., Poggio, T.: Robust Object Recognition with Cortex-like Mechanisms. IEEE PAMI 29, 411–426 (2007)
17. Hammer, B., Gersmann, K.: A Note on the Universal Approximation Capability of Support Vector Machines. Neural Processing Letters 17, 43–53 (2003)
18. Ishikawa, M.: Structural Learning with Forgetting. Neural Netw. 9, 509–521 (1996)
19. Nicolelis, M.A.L., Lebedev, M.A.: Principles of Neural Ensemble Physiology Underlying the Operation of Brain-machine Interfaces. Nature Reviews Neuroscience 10, 530–540 (2009)
20. Simoncelli, E.P., Olshausen, B.A.: Natural Image Statistics and Neural Representation. Annu. Rev. Neurosci. 24, 1193–1216 (2001)
21. Nishimoto, S., Ishida, T., Ohzawa, I.: Receptive Field Properties of Neurons in the Early Visual Cortex Revealed by Local Spectral Reverse Correlation. J. Neurosci. 26, 3269–3280 (2006)
22. Fujiwara, Y., Miyawaki, Y., Kamitani, Y.: Estimating Image Bases for Visual Image Reconstruction from Human Brain Activity. In: Advances in Neural Information Processing Systems, vol. 22, pp. 576–584. MIT Press, Cambridge (2009)
23. Lanckriet, G.R.G., Cristianini, N., Bartlett, P., El Ghaoui, L., Jordan, M.I.: Learning the Kernel Matrix with Semidefinite Programming. J. Mach. Learn. Res. 5, 27–72 (2004)

Adaptive Colour Calibration for Object Tracking under Spatially-Varying Illumination Environments

Heesang Shin, Napoleon H. Reyes, and Andre L. Barczak

Institute of Information and Mathematical Sciences,
Massey University, Auckland, New Zealand
{H.Shin,N.H.Reyes,A.L.Barczak}@massey.ac.nz

Abstract. In the context of a Fuzzy-Genetic system, auto-calibration of colour classifiers, under spatially varying illumination conditions, to produce near perfect object recognition accuracy requires a balancing act for the fitness function. One general approach would be to maximise the true positives while minimising the false positives. This has been found effective in the presence of large amount of noise. However, experiments show that this fitness function needs improvement for cases where there are target colours with similar hues. In this paper, we present an extension to our fuzzy-genetic colour contrast fusion algorithm, now utilising a fitness function that detects clusters of false positives, and limits the search space for finding the properties of the colour classifier. We tested the performance of the auto-calibrated colour classifiers by subjecting them to object recognition tasks in the robot soccer domain, under varying illumination conditions, until we find its limits. It was observed that the accuracy of the object recognition began to degrade, on the average, at illumination settings that are either about three times brighter (starting from 797.4 lux), or two times darker (less than 138 lux) than what it was trained for (average of 285.47 lux). Otherwise, near perfect recognition accuracy is achieved.

Keywords: Colour Classification, Fuzzy Hybrid, Genetic Algorithm.

1 Introduction

Auto-calibration of colour classifiers to discriminate colours of similar hues, to produce near perfect object recognition accuracy still remains a challenge [14]. We investigated several candidate colour classifiers, automatically produced by the Fuzzy Colour Contrast Fusion with Heuristic-Assisted Genetic Algorithm (FCCF-HAGA) [14] for competing colours, such as pink, red and orange, and observed that there are some better candidate colour classifiers that were generated by the same algorithm, but not ranked best by the fitness function. In terms of colour-based object recognition tasks, we found that the fitness function overlooks the best classifier because it does not penalise unwanted ambiguous colour formations. In the light of this problem, we devised a new fitness function that

employs false positive cluster analysis. Using the robot soccer platform [7] as our test bed, our experiments gave us near-perfect object recognition results, within the illumination levels of the training set. In addition, the proposed classifier has proven effective, also with near perfect results for illumination conditions even outside the range of the training set by 73% increase in brightness, in lux units.

2   Review of Related Works

There has been an influx of interest in colour-based object recognition research as technology paved the way for fast image capturing devices to be readily accessible. Pixel-by-pixel classification techniques are usually employed as a colour pixel provides a convenient, constrained and a fixed dimensional space, suitable for supervised classification techniques. It is however identified that spectral information alone would not suffice for solving segmentation problems such as identifying burned forest, roads and golf courses [8,5]. Therefore, a combination of spatial neighbourhood information, as well as arithmetic combination of spectral components would help solve complex segmentation problems [8,15].

Previous works combining Fuzzy Logic and Genetic Algorithms [6,11] for colour calibration usually involves finding the best parameters for the membership functions, using a traditional zero-order Sugeno fuzzy inference system. In this paper, on the contrary, we are using Fuzzy Logic techniques for colour correction purposes and Genetic algorithm for extracting the rules and the properties of the membership functions. Moreover, we employ an unorthodox fuzzy architecture based on the works in [9,10]. There has been researches that considered the spatial relationship between colour classes in a single colour space to discriminate them effectively [1,3]. Colour drifting as a result of changes in the illumination is tracked down to define a cluster of colours as a target colour classifier. In the proposed algorithm, we actually manipulate the colour information to fit into a pie-slice decision region in a 2-D colour space [10]. We also inspected multiple colour spaces to extract the best one for each target colour [14].

Colour resolution reduction has been applied as a preprocessing step for most colour image quantisation problems [16,12]. In [13], it was shown that a GA-based technique for exploring variable colour depths is effective not only for colour quantisation but also for reducing space requirements, and improving colour classification accuracy.

Colour segmentation is still regarded as an open issue and worthy of consideration [4]. Finding the colour patches robustly under changing illuminations is vital to precise robot navigation. In the robot soccer game, the camera is in fact the only source of input from the environment. The complete game strategy depends on the accuracy of robot recognition and heading angle calculations of the vision system. CMVision [2] is one example of a widely used colour classification algorithm based on threshold selection in the YUV colour space.

3   FCCF-HAGA Algorithm

The algorithm is mainly an extension of the works in [13,14]. The hybrid algorithm searches for the best FCCF colour classifier with Variable Colour Depth (VCD) and Multiple Colour Spaces (MCS) using HAGA. The principal contributions of this paper are the introduction of the target colour angle range approximation algorithm and the false positive cluster analysis algorithm. The former reduces the search space of the calibration process, while the latter improves the colour discriminability of competing colours with similar hues.

All these new ideas and algorithms are integrated into a newly-improved FCCF-HAGA, as illustrated in Figure 1.
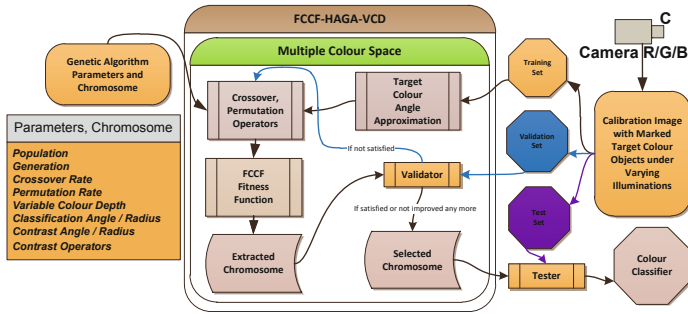
Fig. 1. Architecture Diagram of FCCF-HAGA Colour Classifier Extraction Algorithm

3.1   Target Colour Angle Range Approximation

After investigating multiple colour spaces, we noticed that most target colour pixels are loosely clustered as the hue of target colours usually change smoothly. In the FCCF-HAGA algorithm, finding the optimal pie-slice decision regions and contrast angles are the most important tasks. We can reduce the search space significantly if we can set some limitations on the aforementioned angles. We present a histogram-based approach for estimating the limits of the angles (Algorithm 1). This algorithm builds an angle histogram corresponding to all chromatic target colour pixels. By anchoring from the peak in the histogram, we expand the selection of angles until we find a point that contains less than half the value of the peak. This is done on both left and right directions. This defines the upper and lower bounds of the angle search space.

3.2   False Positive Cluster Analysis Algorithm

From a pool of candidate colour classifiers, representing different colour spaces, extracted using the fitness function in [14], we found that some of the top-ranked classifiers performed less effective than those with slightly lower scores. Further analysis revealed that the fitness function needs to take into account

Algorithm 1. Approximating the Range of Angles for the Target Colour

input : Pixels in target area
output: Lower and upper bound of angle search range

AH[1 to a]: Angle histogram is divided into $\frac{360}{a}$ degrees;
foreach *p ← every target n pixel* **do**
  if *p is chromatic* **then**
    A = Angle of target pixel in colour space;
    b = Histogram position of angle A;
    AH[b] = AH[b] + 1;

AHmax = l = r = Position of maximum value in *AH[1 to a]*;
repeat
    decrease *l*;
    if *l < 1* **then**
      l = a;
until AH[l] *is less than* $\frac{1}{2}$ *of* AH[AHmax] *AND not l* = AHmax ;
repeat
    if *r > a* **then**
      r = 1;
until AH[r] *is less than* $\frac{1}{2}$ *of* AH[AHmax] *AND not r* = AHmax ;
Astart = l$\frac{360}{a}$ = Lower bound of angle search range;
Aend = (r − 1)$\frac{360}{a}$ = Upper bound of angle search range;

the clustering of false positive pixels. To overcome this problem, we devised a new fitness function that penalises the formation of false positive clusters, while maximising the true positives. Eqn. (1) is the new improved fitness function based on [14]. The true positive score was empirically derived to define a function that gradually increases the rewards when the true positive ratio approaches 0.3. It also increases more rapidly when the false positive ratio is lower than 0.1. On the other hand, the false positive score was based on [14], with the penalising factor added. In the worst case scenario, when $k = c$, the false positive score is halved. The final fitness is simply an average of the true positive and false positive scores.

$$x = \frac{true\ positive\ pixels\ count\ within\ the\ target\ area}{total\ pixels\ in\ the\ target\ area}$$

$$y = \frac{false\ positive\ pixels\ outside\ target\ area}{total\ pixels\ outside\ the\ target\ area}$$

$$true\ positive\ score = \frac{1 - \frac{1}{1+e^{-75(x-0.05)}}}{1 + e^{-10(y-0.4)}}$$

$$z = average\ number\ of\ pixels\ within\ a\ cluster\ of\ true\ positives$$

$$c = number\ of\ false\ positive\ clusters(FPC)$$

$$k = number\ of\ FPC\ whose\ total\ pixel\ count\ is\ > \frac{1}{2}z$$

$$false\ positive\ score = \left(\frac{(1 - \frac{k}{c})}{2} + 0.5 \right) \frac{1}{e^{-7(y-0.7)}}$$

$$fitness = \frac{true\ positive\ score + false\ positive\ score}{2}$$

(1)

4   Experiments

We have conducted two main experiments to characterise the efficacy of the algorithms. The first one is to extract the best colour classifiers for each of the 5 competing algorithms. Secondly, we have utilised the extracted classifiers to evaluate their efficacy in robot tracking.

4.1   Experiments Setup

The experiment was conducted in a controlled environment, meeting the standards of the FIRA robot soccer platform as shown in Figure 2. Multiple sets of adjustable lightings are used to produce an exploratory space with spatially varying illumination conditions. Using a Minolta CL-200 Chroma Metre, the illumination of the surface of the playing field was quantified in terms of the CIE 1931 X, Y, Z tristimulus values. The points of measurements were taken from 9 different sections in the playing field, as illustrated in Figure 3. An illumination distribution described in Table 1. A Prosilica GC-650 digital camera, with 4.5mm / F1.4 lens was used in all of the experiments. It was configured with an exposure time setting of 20ms.

Fig. 2.   Experiment Environment
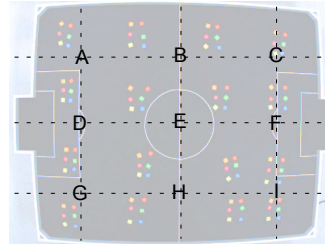
Fig. 3.   Illumination Measurement Points

4.2   Automated Colour Classifier Extraction

Four main algorithms were used to extract the best colour classifiers. Two static images, with spatially varying illumination condition (Figure 4), were used for training and validation sets by the classifiers. In addition, as depicted in Figure 4, there are 6 target colours to be tracked (green, light blue, orange, pink, red and yellow). For each of the algorithms, we have extracted 6 sets of colour classifiers corresponding to the target colours. We have labelled these extracted sets of colour classifiers as follows:

– FCCF-HAGA-A : extracted using the newly proposed FCCF-HAGA algorithm.
– FCCF-HAGA-B : extracted using the former FCCF-HAGA algorithm described in [14].
– FCCF-HAGA-C : extracted using FCCF-HAGA-B but ranked manually with the aid of a method discussed in 3.2.
– CMVision-GA : extracted using an algorithm described in [2], and calibrated by a Genetic Algorithm with a fitness function defined in Eqn. (1).

(a) Training Set   (b) Validation Set
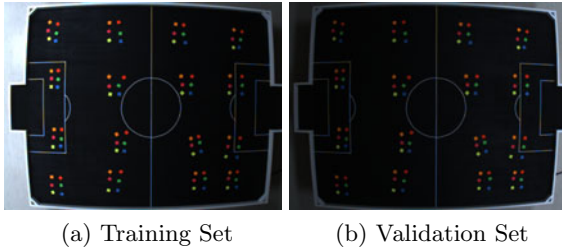
Fig. 4. Robot Soccer Playing Field under Spatially Varying Illumination

4.3   Robot Tracking

In order to compare the performance of the different algorithms to be analysed, we have recorded several videos of a moving robot, using an overhead camera, while it traverses the standard robot soccer playing field. Six videos serve as the test sets, designated as Video (A) to (F). Video (A) and Video (B) were taken at the same lighting conditions as the training and validation sets. Table 1 show the detailed illumination quantification of the videos used for testing. Some frames were skipped during the recording process due to the limitations our hardware.

Table 1. Illumination Quantification of the Scenes in the Videos

Measured	Training Setting			Validation Setting			Test Setting														
	Video (A)			Video (B)			Video (C)			Video (D)			Video (E)			Video (F)		
Point	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z
A	496.3	500.1	250.7	251.2	252.1	125.5	250.7	251.5	151.2	27.5	27.3	10.6	811.1	816.1	400.6	1009	1009	522
B	311.3	312.9	154.7	359.0	361.1	182.2	310.6	312	187.8	38.6	38.4	15.6	780.5	785	384.4	1086	1091	568
C	195.2	194.5	195.7	511.3	514.4	262.0	248.9	249.3	149.7	72.2	71.9	31.8	862.2	865.8	426.9	1060	1065	548
D	404.4	406.7	202.2	259.8	260.9	129.9	261.6	262.6	157.3	34.7	34.4	13.5	781.5	785.8	383.9	1044	1051	545
E	205.8	205.2	99.3	364.8	366.7	185.0	329.1	331.5	199.2	47.3	46.7	19.5	766.5	770.7	376.6	1095	1103	576
F	182.6	182.7	89.3	516.5	520.7	265.8	261.7	263.7	157.6	78.1	77.4	34.8	848.5	851.6	420.3	1116	1125	582
G	413.8	416	206.8	257.3	256.3	127.9	228.9	230.2	137.1	36.5	36.3	14.3	728.5	732.4	356.6	955.2	960.3	492.8
H	210.8	210.9	102.5	369.1	369.0	187.0	287.5	288.8	173.2	49	48.4	19.6	736.4	738.1	360.6	1008	1012	525
I	138.3	138.4	66.6	516.0	519.2	266.2	226.4	226.7	135.1	77.7	76.5	33.2	828.8	831.1	410	1056	1058	545
Min	138.3	138.4	66.6	251.2	252.1	125.5	226.4	226.7	135.1	27.5	27.3	10.6	728.5	732.4	356.6	955.2	960.3	492.8
Max	496.3	500.1	250.7	516.5	520.7	266.2	329.1	331.5	199.2	78.1	77.4	34.8	862.2	865.8	426.9	1116	1125	582
Average	284.28	285.27	151.98	378.33	380.04	192.39	267.27	268.48	160.91	51.29	50.81	21.43	793.78	797.40	391.10	1047.69	1052.70	544.87

5   Results and Discussion

5.1   Colour Classification Results

For each target colour, the automatic colour classifier extraction system was run until it meets either a fitness value of 0.95 or a convergence rate of 0.99. The termination condition is based on a review of the fittest chromosomes from the last 20 generations. The details of the extracted colour classifiers are shown in Table 2. It can be seen that the best classifiers automatically selected come from different colour spaces. The colour depth indicates how many bits were used to represent each of the colour channels. On the other hand, the contrast rules indicate the fuzzy contrast operator and level of operation used. For example, (-1, 2, -1) means apply the fuzzy degradation operation once on the red and blue channels, while fuzzy enhancing the green channel twice. The actual colour

classification results for the 6 target colours are depicted in Figure 5. It is evident from the figure that there are no longer noticeable concentration of false positive clusters.

Table 2. FCCF-HAGA Extracted Colour Classifiers

Classifier Name	Fitness Score	Colour Space	Colour Depth (R,G,B)	Contrast Rules (R,G,B)	Classifier Name	Fitness Score	Colour Space	Colour Depth (R,G,B)	Contrast Rules (R,G,B)
Green	0.915	AB	6,6,8	0,0,0	Light Blue	0.81	RG	8,6,5	-1,0,0
Orange	0.894	C1C2	8,8,5	0,1,0	Pink	0.923	CM	5,7,7	-1,0,0
Red	0.921	C1C2	7,7,7	-1,2,-1	Yellow	0.908	AB	7,6,7	1,3,1

(a) Green   (b) Light Blue   (c) Orange   (d) Pink   (e) Red   (f) Yellow
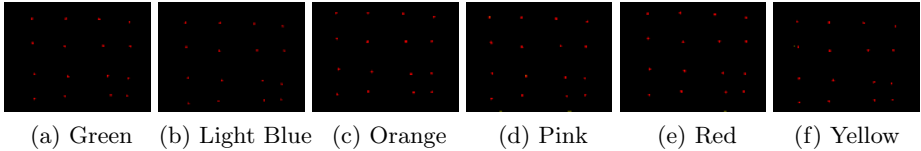
Fig. 5. Colour Classification Results for the 6 Colour Targets. Red Pixels: True Positives; Yellow Pixels: False Positives; Green Pixels: True Negatives.

5.2   Colour Classification Algorithm Comparisons

Table 3 depicts the detailed comparisons of the colour classifiers systems described in 4.2. The scores were calculated by the newly introduced fitness function, except for the scores enclosed in parenthesis, which are calculated by the fitness function in [14]. From the results in FCCF-HAGA-B, we can see that most of the new fitness values are much lower than that of the original ones (parenthesised ones). This is due to the penalty imposed on the formation of false positive clusters. It is worth-noting that with the introduction of the new fitness function on FCCF-HAGA-B gave a boost on it's performance, also agreeing with the manually selected classifiers of FCCF-HAGA-C. However, when the new fitness function was employed in combination with the thresholding classifier algorithm of CMVision-GA, it still failed to isolate red from pink and orange, even after many attempts (see Figure 6). This clearly indicates the limitations of the simple thresholding algorithm of CMVision-GA. Using an Intel 2.8 GHz i7 CPU, the average time to perform colour extraction per colour are as follows: 353sec. (FCCF-HAGA-A), 542sec.(FCCF-HAGA-B), 431sec.(FCCF-HAGA-C) and 614sec.(CMVision-GA) seconds respectively.

5.3   Robot Tracking Test Results

We fed the video frames into our robot tracking algorithm, with the extracted colour classifier sets found in 5.2. The tracking algorithm involves Green, Light Blue, Pink and Red classifiers only to track the robot, therefore Orange and Yellow colour targets were not classified anymore in the process. We measured

the tracking accuracy by considering the calculated position and heading angle of the robot as compared to standard values based on human visual judgement. We sampled 100 frames from the videos (Table 4) to accomplish this. The error corresponding to the calculated positions and heading of the robot as it traverses the field is based on a Euclidean distance and heading angle measurement. Out of 100 frames, for FCCF-HAGA-A, within a tolerance of 2.5 pixel units, the accuracy is 99%, with a standard deviation of 0.676 and heading angle within a tolerance of ± 7 degrees, the accuracy is 92%, with a standard deviation of 3.04. The rest of the results is as follows: FCCF-HAGA-B: 86% accuracy, std. deviation of 1.703 in position and 87% accuracy, std. deviation of 27.75 in heading angle; FCCF-HAGA-C: 99% accuracy, std. deviation of 0.682 in position and 88% accuracy, std. deviation of 3.31 in heading angle; CMV-GA: 93% accuracy, std. deviation of 0.823 and 91% accuracy, std. deviation of 3.37 in heading angle; Table 4 details the robot tracking results, with an indication of the general illumination condition measured in lux units. The percentage error of the calculated robot's position and heading angle is also reported. Figure 7 shows an example of the robot tracking results, as the robot traverses a darker to brighter illuminated area. It is worth noting that the object tracking algorithm employed in the experiments is able to recover from misclassifications of colour patches. It tracks 4 colour patches depicting a robot. Because of this, it is not obvious to see that CMV-GA actually fails to distinguish pink from red, but the tracking algorithm still finds the robot. The recorded average processing time per frame was 7 ms (without the image acquisition time). All experiment results (videos) are available at http://cvlab.massey.ac.nz.

Table 3. Colour Classifier Comparisons. Score: Fitness values, scores in parenthesis correspond to the old fitness function score; TP: number of true positive pixels; FP: number of false positive pixels.

Classifier	FCCF-HAGA (A)			FCCF-HAGA (B)			FCCF-HAGA (C)			CMVision-GA		
	Score	TP	FP	Score	TP	FP	Score	TP	FP	Score	TP	FP
Green	0.915	946	547	0.923 (0.923)	968	1126	0.923 (0.923)	968	1126	0.89	899	1159
Light Blue	0.81	800	1000	0.653 (0.891)	926	2428	0.653 (0.891)	926	2428	0.691	980	5644
Orange	0.894	1020	1234	0.695 (0.921)	1101	2484	0.867 (0.866)	996	874	0.744	806	478
Pink	0.923	996	1529	0.684 (0.926)	1000	1021	0.684 (0.926)	1000	1021	0.872	887	654
Red	0.921	1126	1487	0.683 (0.921)	1146	2885	0.915 (0.915)	1111	1579	0.729	1145	5174
Yellow	0.908	856	1433	0.676 (0.916)	875	1752	0.676 (0.916)	875	1752	0.86	775	606

Table 4. Comparisons of Robot Tracking Performance Utilising the Colour Classifier Sets

Classifier Sets	Video (A) Average 285.47lux (Dim) Tracking Rate	Video (B) Average 380.17lux (Average) Tracking Rate	Video (C) Average 268.48lux (Dimmer) Tracking Rate	Video (D) Average 51.29lux (Darkest) Tracking Rate	Video (E) Average 797.4lux (Bright) Tracking Rate	Video (F) Average 1052.7lux (Brightest) Tracking Rate
FCCF-HAGA (A)	100.00%	100.00%	100.00%	0.31%	100.00%	6.87%
FCCF-HAGA (B)	100.00%	99.90%	100.00%	41.82%	99.81%	35.48%
FCCF-HAGA (C)	100.00%	99.90%	100.00%	26.73%	99.81%	29.27%
CMV-GA	98.69%	100.00%	100.00%	0.31%	99.42%	40.13%

(a) Pop. 100   (b) Pop. 150   (c) Pop. 200   (d) Pop. 300
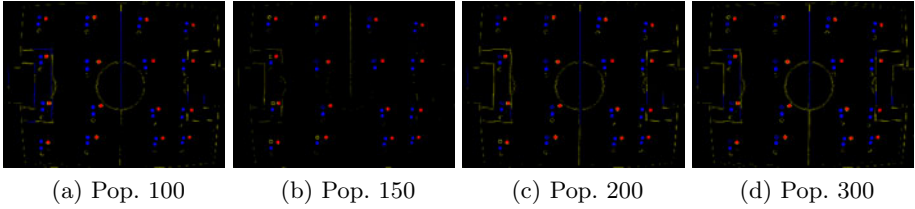
Fig. 6. Colour Classification Results for Red using CMVision-GA Classifier with Different Population Parameters. Red Pixels: True Positives; Yellow Pixels: False Positives; Green Pixels: True Negatives; Blue Pixels: False Positive Clusters.

Fig. 7. Robot Traversal of a Spatially Varying Illumination Field. (Above) Consolidated Sequences of Robot's Motion, (Below) Robot Tracking Results. White circle denotes the recognised robot and the white line segment denotes the calculated robot heading angle.

6 Conclusions

We have devised a new fitness function for a Heuristic-Assisted Genetic Algorithm that works with the Fuzzy Colour Contrast Fusion algorithm [14]. The new fitness function penalises the formation of false positive clusters as we have observed that robot tracking tasks are adversely affected by them. Thorough experiments using the robot soccer domain were conducted to compare the results of the proposed system against the algorithms described in [14], [2] and manually selected colour classifiers. The proposed system was capable of isolating all target colours effectively, as compared to the thresholding algorithm of CMVision which failed isolating red from pink and orange. The colour classifiers were trained on static images, but were tested on recorded video sequences of a moving robot traversing a field with spatially varying illumination conditions. The results of the experiments demonstrated that the new system achieved near-perfect robot tracking results, within the range between 138.4 lux and 865.8 lux. These test scenes were two times darker and three times brighter than the average illumination level for which the system was trained for.

References

1. Anzani, F., Bosisio, D., Matteucci, M., Sorrenti, D.G.: On-Line Color Calibration in Non-Stationary Environments. In: Bredenfeld, A., Jacoff, A., Noda, I., Takahashi, Y. (eds.) RoboCup 2005. LNCS (LNAI), vol. 4020, pp. 396–407. Springer, Heidelberg (2006)
2. Bruce, J., Balch, T., Veloso, M.: Fast and inexpensive color image segmentation for interactive robots. In: Proceedings of IROS 2000, pp. 2061–2066 (2000)
3. Guerrero, P.E., Ruiz-del-Solar, J., Fredes, J., Palma-Amestoy, R.: Automatic On-Line Color Calibration using Class-Relative Color Spaces. In: Visser, U., Ribeiro, F., Ohashi, T., Dellaert, F. (eds.) RoboCup 2007: Robot Soccer World Cup XI. LNCS (LNAI), vol. 5001, pp. 246–253. Springer, Heidelberg (2008)
4. Gurzoni, J., Martins, M., Tonidandel, F., Bianchi, R.: On the construction of a robocup small size league team. Journal of the Brazilian Computer Society 17, 69–82 (2011)
5. Harvey, N., Theiler, J., Brumby, S., Perkins, S., Szymanski, J., Bloch, J., Porter, R., Galassi, M., Young, A.: Comparison of genie and conventional supervised classifiers for multispectral image feature extraction. IEEE Transactions on Geoscience and Remote Sensing 40(2), 393–404 (2002)
6. Kashanipour, A., Milani, N.S., Kashanipour, A.R., Eghrary, H.H.: Robust color classification using fuzzy rule-based particle swarm optimization. In: Li, D., Deng, G. (eds.) CISP 2008: Proceedings of First International Congress on Image And Signal Processing, China, May 27-30, vol. 2, pp. 110–114. Tianjin Univ. Technol., IEEE Computer Soc., Sanya, Peoples R China (2008)
7. Kim, J.H., Seow, K.T.: Soccer Robotics. Springer, Heidelberg (2004)
8. Perkins, S.J., Theiler, J., Brumby, S.P., Harvey, N.R., Porter, R.B., Szymanski, J.J., Bloch, J.J.: GENIE: a hybrid genetic algorithm for feature classification in multispectral images. In: Bosacchi, B., Fogel, D.B., Bezdek, J.C. (eds.) SPIE Conference Series, vol. 4120, pp. 52–62 (2000)
9. Reyes, N.H., Dadios, P.E.: Dynamic color object recognition using fuzzy logic. Journal of Advanced Computational Intelligence and Intelligent Informatics 8, 29–38 (2004)
10. Reyes, N.H., Messom, C.: Identifying colour objects with fuzzy colour contrast fusion. In: 3rd International Conference on Computational Intelligence, Robotics and Autonomous Systems, and FIRA RoboWorld Congress (2005)
11. Sakurai, M., Kurihara, Y., Karasawa, S.: Color classification using fuzzy inference and genetic algorithm. In: Proceedings of the Third IEEE Conference on IEEE World Congress on Computational Intelligence, Fuzzy Systems, June 26-29, vol. 3, pp. 1975–1978 (1994)
12. Scheunders, P.: A genetic c-means clustering algorithm applied to color image quantization. Pattern Recognition 30(6), 859–866 (1997)
13. Shin, H., Reyes, N.: Finding near optimum colour classifiers: genetic algorithm-assisted fuzzy colour contrast fusion using variable colour depth. Memetic Computing Journal, 1–18 (2009)
14. Shin, H., Reyes, N.H., Barczak, A.L.: A Hybrid Fuzzy-Genetic Colour Classification System with Best Colour Space Selection under Dynamically-Changing Illumination. In: Wong, K.W., Mendis, B.S.U., Bouzerdoum, A. (eds.) ICONIP 2010. LNCS, vol. 6444, pp. 291–299. Springer, Heidelberg (2010)

15. Shin, H., Reyes, N.H., Barczak, A.L., Chan, C.S.: Colour Object Classification Using the Fusion of Visible and Near-Infrared Spectra. In: Zhang, B.-T., Orgun, M.A. (eds.) PRICAI 2010. LNCS, vol. 6230, pp. 498–509. Springer, Heidelberg (2010)
16. Stachowicz, M.S., Lemke, D.: Color recognition. In: Proceedings of the 22nd International Conference on Information Technology Interfaces (ITI 2000), pp. 329–334 (2000)

Analog-Digital Circuit for Motion Detection Based on Vertebrate Retina and Its Application to Mobile Robot

Kimihiro Nishio and Taiki Yasuda

Department of Electrical and Electronic Engineering,
Tsuyama National College of Technology,
624-1, Numa, Tsuyama, Okayama, 708-8509, Japan
nishio@tsuyama-ct.ac.jp

Abstract. We proposed in this study the simple analog-digital circuits for detecting motion direction based on information processing of the vertebrate retina. The array of the circuits was applied to the mobile robot. The test circuit was fabricated by discrete metal oxide semiconductor (MOS) transistors on the breadboard. The measured results of the test circuit showed that the unit circuit can output the motion signal. The motion sensor for detecting the movement direction constructed with array of the unit circuits was connected with the microcomputer introduced in the mobile robot. It was clarified that the proposed circuits can control the mobile robot.

Keywords: analog circuit, digital circuit, mobile robot, vision chip.

1 Introduction

High speed and compact image processing system is needed for robotics vision and other systems. However, it is difficult to realize the high speed processing system. The typical image processing system is constructed with the image sensor and Neumann-type computer. Since the computer process the information in time sequential way, it is difficult to realize the high speed processing system by using the typical image processing system. The vertebrate retina is preprocessor for the image processing in the brain. The retina has the functions such as the edge detection and the generation of the motion signal. The retina can perform the high speed processing since the nerve cells in the retina perform information processing in parallel.

Many researchers proposed the motion detection circuit by mimicking the information processing of the retina and brain [1]-[8]. The circuits are characterized by high speed processing. Researchers tried to apply these circuits to the system such as the robot, target tracking and other systems. However, these circuits have a problem of incorrect operation by device mismatches due to use the analog technology. And, there is also problem of low resolution. Thus, it is difficult to use these analog circuits to the application systems.

Digital circuits for motion detection were proposed based on the biological vision system [2],[9]. There is no problem of incorrect operation. However, there is the

problem of the complex structure by using digital technology. If the simple digital circuit for motion detection is proposed, it is able to apply it to many systems.

Such motion detection circuits are needed for the robotics vision and other application systems. It is necessary to propose the application systems using such circuits.

In this study, we tried to propose the simple analog-digital circuits for detecting the motion direction by mimicking the vertebrate retina. The measured results of the test circuit showed that the unit circuit can generate the motion signal. The motion sensor for detecting the movement direction constructed with array of the unit circuits was connected with the microcomputer introduced in the mobile robot. The measured results showed that the proposed circuits can control the mobile robot.

2   Motion Detection Circuit

2.1   Motion Detection Model

Figure 1 shows the model for generating a motion signal based on information processing of the vertebrate retina [6],[7]. Figure 1(a) shows the relationship between one photoreceptor P (input part) shown by a gray square and an edge of the projected image. The edge moves toward the right hand side with a constant velocity v.

Figure 1(b) shows the transient response of the model for generating a motion signal. When the edge of the object moves on the P, the edge signal V_a is generated, i.e. V_a becomes the constant value V_{con}. After time t_d, the signal V_b is generated, i.e. V_b becomes V_{con}. The sum of V_a and V_b is the output signal V_{o1} of the model. These pulsed signals are motion signals. In this study, the positive pulsed signal is the motion signal. The signal V_{out} is generated when the object moves on the P.

In the retina, the photoreceptors are arrayed in one- and two-dimensionally [6],[7]. As the example, the one-dimensional array of photoreceptors is shown in Fig. 1(c). This array detects the movement direction of the object. Each signal from the cell is generated when the object moves on P. Since the signal from each cell is integrated, the output signal is large when the object moves on the right or left hand side. By the array of the cells, the movement direction of the object can be detected.

2.2   Unit Circuit

Figure 2 shows the unit analog-digital circuit for motion detection based on the model in Figs. 1(a) and (b). The unit circuit is constructed with the edge detection circuit [6]-[8] in Fig. 2(a) and the circuit for generating the motion signal in Fig. 2(b). The circuit in Fig. 2(a) is connected with the circuit in Fig. 2(b). The edge detection circuit is realized by the analog technology. The circuit for generating the motion signal is realized by the digital technology.

The NOT circuit consists of two metal oxide semiconductor (MOS) transistors. The NOR circuit consists of four MOS transistors. The circuit is constructed with six MOS transistors. The proposed circuit is simple structure as compared with the previous circuit [1]-[9].
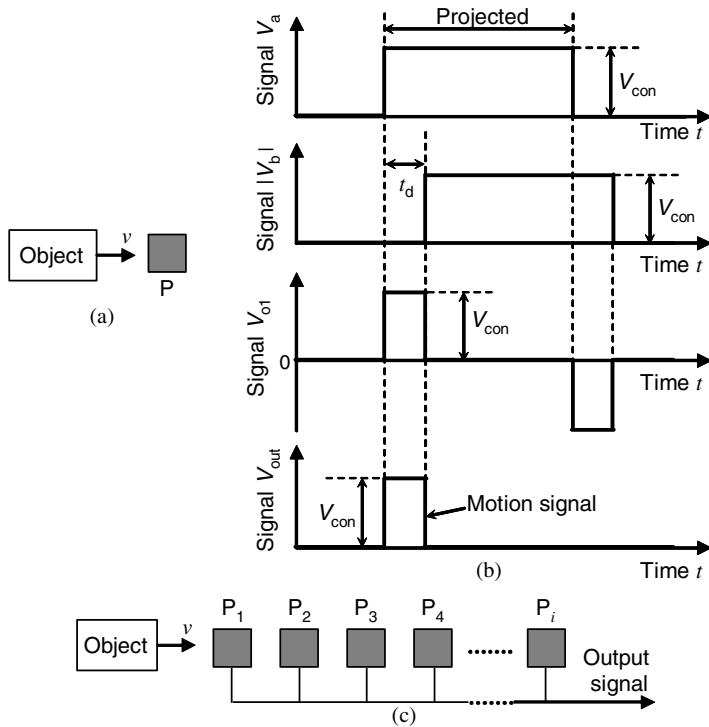
Fig. 1. Model for generating a motion signal. (a) Relationship between a photoreceptor and the edge of the object. (b) Transient response of the model. (c) One-dimensional array of photoreceptors.
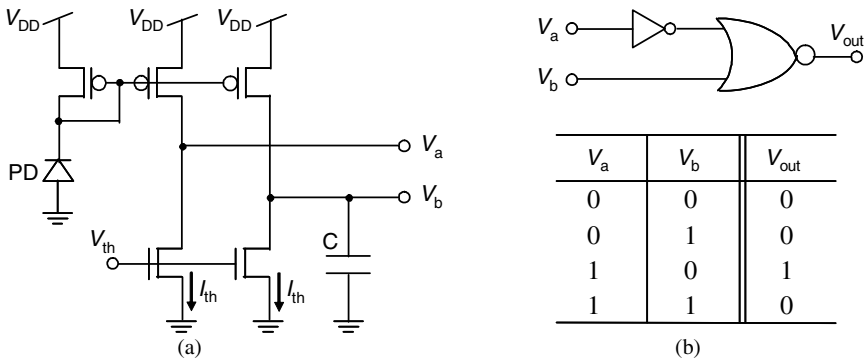
Fig. 2. Unit circuit for motion detection. (a) Edge detector. (b) Circuit for generating the motion signal.

In this circuit, the constant voltage V_{th} is set. The constant current I_{th} is generated by V_{th}. The photodiode PD corresponds to the photoreceptor P in Fig. 1. The

voltages V_a, V_b and V_{out} in Fig. 2(b) correspond to the signals V_a, V_b and V_{out} in Fig. 1, respectively. When the object is projected on PD, V_a becomes about the supply voltage V_{DD} (= V_{con}). V_b becomes about V_{DD} (= V_{con}) after the time t_d with capacitor C. As shown in the truth table, the output voltage V_{out} showed V_{DD} for t_d. Thus, V_{out} showed the pulsed voltage as well as the model in Fig. 1 when the edge of the object moves on PD.

There is the problem of the incorrect operation by device mismatches because previous circuits are realized by only analog technology. The proposed circuit does not have problem of incorrect operation since the proposed circuit is realized by the digital technology.

3   Connection with Motion Sensor and Mobile Robot

It is necessary for robotics vision to detect the motion of the object in real time. We tried to connect the motion detection circuits with the mobile robot. The microcomputer is introduced in the mobile robot.

Figure 3 shows the connection with the motion sensor and the microcomputer in the robot. The array of the unit motion detection circuits in Fig. 2 is utilized as the
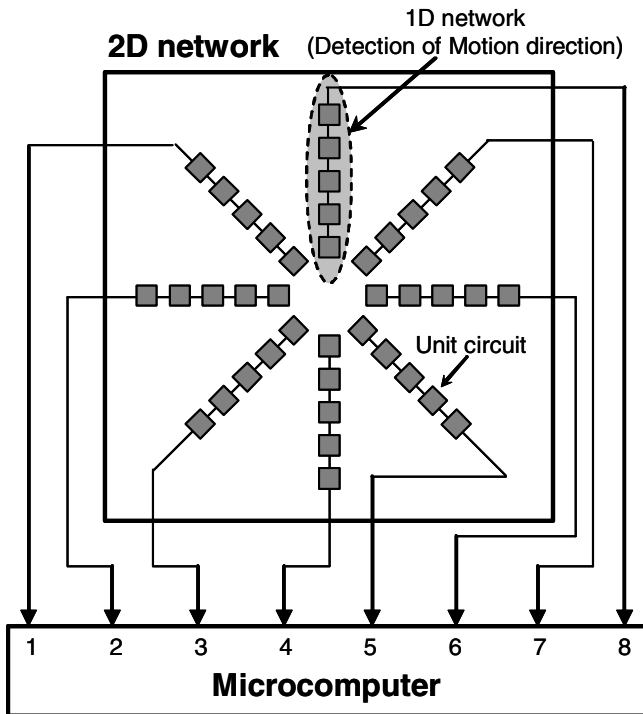
Fig. 3. The connection with the motion sensor and the microcomputer in the robot

motion sensor. The function of the one-dimensional array is equal to that in Fig. 1(c). Each array can detect the motion direction by using the output voltage V_1-V_8. Since it assumed that the sensor can capture the object in the center of the sensor by using the function of the target tracking [10],[11], we use such array in Fig. 3 i.e. the array in the radial pattern. It is able to detect the motion direction by using the array of the unit circuits.

   The microcomputer is introduced in the mobile robot. Each motion signal V_1-V_8 is input to the input ports (node 1-8) of the microcomputer. The mobile robot is controlled by using the input signals and the robot works by the program. It is able to control the mobile robot by using the signals of the motion sensor.

4   Experimental Results

Figure 4 shows the photograph of the fabricated system. The test circuit of the motion detection circuit in Fig. 2 was fabricated and measured. The analog part of the test circuit was fabricated on the breadboard with the discrete MOS transistors (nMOS: 2SK1398, pMOS: 2SJ184, NEC). The digital part was fabricated by the field programmable gate array (FPGA). V_{th} and V_{DD} were set to 1.6 V and 5 V, respectively.   C was set to 4.7 μF.
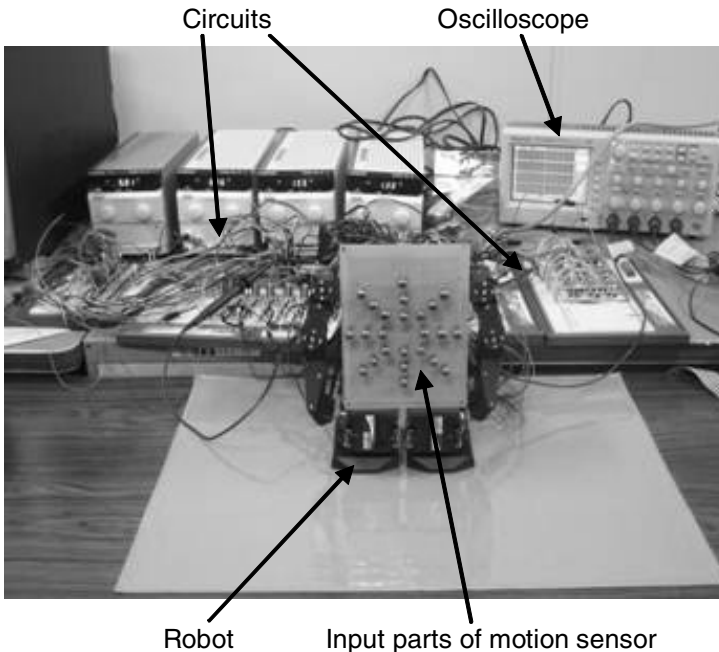
Fig. 4. Photograph of the fabricated system
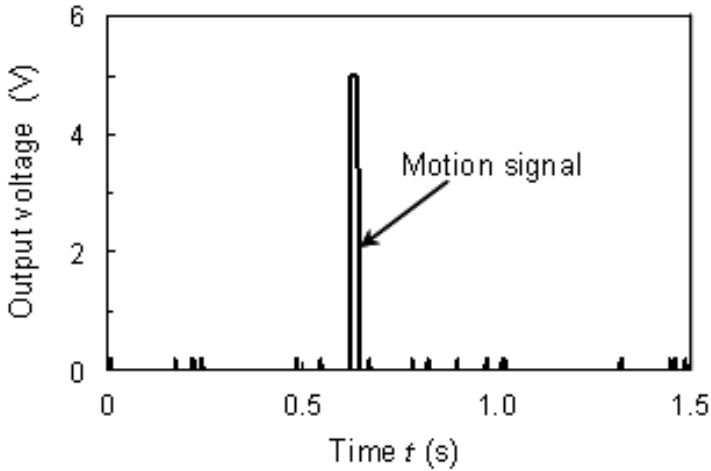
Fig. 5. Measured results of the test circuit

The system was constructed with the motion sensor for detecting the motion direction constructed with the array of the proposed circuits and the mobile robot (ROBONOVA-I, Hitec Multiplex Japan).   The microcomputer (MR-C3024) is introduced in the robot.   In this experiment, we programmed that the robot raises its hands to the movement direction.

In this measurement, the light was provided as the object for the checking of the operation of the unit circuit. Figure 5 shows the measured results of the test circuit. When the object moves on PD, the circuit output the pulsed voltage as well as the model in Fig 1(b). The unit circuit can generate the signal in all measurement. The previous circuit does not often generate the signal by device mismatches. Thus, the unit circuit can generate the motion signal although the structure of the circuit is simple.

Figure 6 shows the measured results. In this measurement, the light was also provided as the object. The robot raised its hands to the movement direction when the light moved on the motion sensor. The robot could operate by using the motion signal. Thus, it was clarified that the array of the motion detection circuit can control the mobile robot.

In this measurement, the real image was not provided since the number of unit circuits constructed with the array is small. The array of the edge detection circuits utilized to the unit circuit can generate the edge signal even if the real image is provided on the array [12],[13]. Thus, the motion sensor can detect the motion direction although the real image is provided.
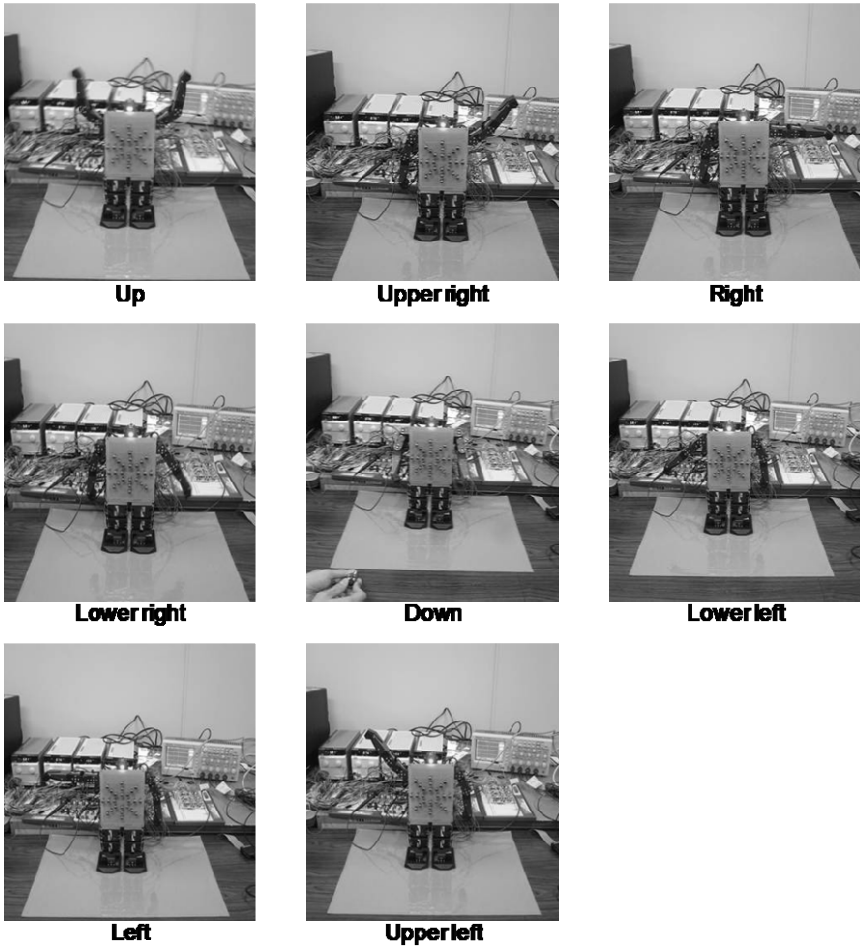
Fig. 6. Measured results of the fabricated system

5   Conclusion

The analog-digital circuit for motion detection was proposed based on the vertebrate retina. The circuit for generating the motion signal is constructed with one NOT circuit and one NOR circuit. The proposed circuit is simple structure.   The test circuit was fabricated by discrete metal oxide semiconductor (MOS) transistors on the breadboard. The measured results of the test circuit showed that the proposed circuit can output the motion signal. The motion sensor for detecting the movement direction constructed with array of the proposed circuits was applied to the mobile robot.   The motion sensor was connected with the microcomputer introduced in the mobile robot. It was clarified that the array of the proposed circuits can control the mobile robot. In the future, it is able to use the proposed motion sensor as the robotics vision.

References

1. Mead, C.A.: Analog VLSI and Neural Systems. Addison Wesley, Reading (1989)
2. Moini, A.: Vision Chips. Kluwer Academic, Norwell (1999)
3. Liu, S.C., Viretta, A.U.: Fly-Like Visuomotor Responses of a Robot Using a VLSI Motion-Sensitive Chips. Biological Cybernetics 85, 449–457 (2001)
4. Liu, S.C.: A Neuromorphic a VLSI Model of Global Motion Processing in the Fly. IEEE Trans. on Circuit and Systems II 47, 1458–1467 (2000)
5. Asai, T., Ohtani, M., Yonezu, H.: Analog MOS Circuits for Motion Detection Based on Correlation Neural Networks. Jpn. J. Appl. Phys. 38, 2256–2261 (1999)
6. Yamada, H., Miyashita, T., Ohtani, M., Nishio, K., Yonezu, H., Furukawa, Y.: Signal Formation of Image-Edge Motion Based on Biological Retinal Networks and Implementation into an Analog Metal-Oxide-Silicon Circuit. Optical Review 8, 336–342 (2001)
7. Nishio, K., Matsuzaka, K., Yonezu, H.: Simple Analog Complementary Metal Oxide Semiconductor Circuit for Generating Motion Signal. Optical Review 14, 282–289 (2007)
8. Nishio, K., Yonezu, H., Kariyawasam, A.B., Yoshikawa, Y., Sawa, S., Furukawa, Y.: Analog Integrated Circuit for Motion Detection against Moving Background Based on the Insect Visual System. Optical Review 11, 24–33 (2004)
9. Boahen, K.A.: Retinomorphic Chips that see Quadruple Images. In: 7th International Conference on Microelectronics for Neural, Fuzzy, and Bio-Inspired Systems (MicroNeuro 1999), pp. 12–20 (1999)
10. Nishio, K., Matsuzaka, K.: Target Tracking System Using Analog Circuit for Motion Detection. In: 2008 IEEE International Conference on Mechatronics and Automation (ICMA 2008), vol. TC3-5 (2008)
11. Nishio, K., Matsuzaka, K.: Target Tracking System Using Analog Motion Detection Circuit Based on Biological Vision System. Far East Journal of Electronics and Communications 3, 111–124 (2009)
12. Nishio, K., Nakahara, H.: Simple Analog Metal Oxide Semiconductor Circuit for Edge Detection. In: The International Conference on Electrical Engineering 2008 (2008)
13. Nishio, K.: Simple Analog Edge Detection Circuit Based on Vertebrate Retina. Optical Review 16, 521–525 (2009)

Spatial Finite Non-gaussian Mixture for Color Image Segmentation

Ali Sefidpour and Nizar Bouguila

Concordia Institute for Information Systems Engineering, Concordia University,
Montreal, Canada, Qc, H3G 2W1
al_sef@encs.concordia.ca, bouguila@ciise.concordia.ca

Abstract. A new color image segmentation algorithm based on the integration of spatial information into finite generalized Dirichlet mixture models is presented. The integration of spatial information is done via the consideration of image pixels neighborhoods. The segmentation model presented is learned using maximum likelihood estimation within an expectation maximization (EM) optimization framework. The obtained results, evaluated quantitatively, using real images are very encouraging and are better than those obtained using similar approaches.

Keywords: Image segmentation, mixture models, spatial information.

1 Introduction

Image segmentation has received a great deal of attention in the image processing literature. This is an important first step for a variety of image analysis and computer vision tasks such as content-based image retrieval [1]. The main goal is to divide up images into homogeneous regions which can be viewed essentially as an unsupervised learning problem [2,3]. The application of statistical models to the image segmentation problem has been a topic of much interest and several approaches and methodologies have been proposed. In particular, finite mixture models have received a lot of attention because they offer an efficient approach to the unsupervised learning (or clustering) problem [4,5]. The direct application of finite mixture models suffer, however, from a well-known deficiency namely the lack of the spatial information (i.e. the spatial relations between the pixels to cluster are generally not taken into account).

There is a vast body of literature on various spatial integration models for the image segmentation problem. For instance, the spatial knowledge has been used in [6] for the segmentation of Landsat images, in [7] for the segmentation of medical images, and in [8] for the segmentation of color textures. Attempts to overcome the lack of spatial information have chiefly centered on the consideration of pixel's neighborhood during the segmentation process [9,10]). For instance, when a finite mixture model is considered for segmentation, it is reasonable to assume that two neighbors will be in the same region as done in [11]. The work in [11] has been based on the assumption that the image regions follow Gaussian distributions which is actually very restrictive as previously shown by

several studies (see, for instance, [12,5,13]). Thus, the goal of this paper is to propose an alternative statistical framework where generalized Dirichlet (GD) distributions, which offer both flexibility and ease of use, are considered. Within the proposed framework, the spatial information is integrated and used as domain knowledge prior information to determine the accurate number of image regions during the segmentation process. In the following section, we first discuss our segmentation statistical framework and then a complete learning approach to estimate its parameters is given. Section 3 is devoted to the presentation of our experimental results. Finally a conclusion is given and future research directions are proposed in Section 4.

2   The Segmentation Model

Let $\mathcal{X}$ be an image represented by a set of pixels $\mathcal{X} = \{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N\}$ where each pixel is denoted by a random vector $\boldsymbol{X}_n = (X_{n1}, \ldots, X_{nD})$ and N is the number of pixels. Now if the random vector $\boldsymbol{X}$ follows a GD distribution with positive parameters $\boldsymbol{\alpha} = (\alpha_1, \beta_1, \ldots, \alpha_D, \beta_D)$, the joint density function is given by [14,15,5]: $p(\boldsymbol{X}|\boldsymbol{\alpha}) = \prod_{d=1}^{D} \frac{\Gamma(\alpha_d+\beta_d)}{\Gamma(\alpha_d)\Gamma(\beta_d)} X_d^{\alpha_d-1} \left(1 - \sum_{i=1}^{d} X_i\right)^{\gamma_d}$, where $\sum_{d=1}^{D} X_d <$ 1 and $0 < X_d < 1$ for $d = 1, \ldots, D$ and $\gamma_d = \beta_d - \alpha_{d+1} - \beta_{d+1}$ for $d = 1, \ldots, D-1$ and $\gamma_d = \beta_d - 1$. Generally, an image is composed of different regions. Thus, it is appropriate to describe it by a finite GD mixture model with M components: $p(\boldsymbol{X}|\Theta) = \sum_{j=1}^{M} P_j p(\boldsymbol{X}|\boldsymbol{\theta}_j)$, where $\{P_j\}$ are the mixing proportions which are positive and sum to one, $p(\boldsymbol{X}|\boldsymbol{\theta}_j)$ is the GD distribution, $\boldsymbol{\theta}_j = (\alpha_{j1}, \beta_{j1}, \ldots, \alpha_{jD}, \beta_{jD})$, and $\Theta = (P_1, \ldots, P_M, \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M)$ is the set of all mixture parameters.

Segmentation can be viewed as the spatially coherent clustering of image features (e.g, color, texture), yet it is clear that a mixture model does not take into account the spatial information. In order to overcome this problem, we adopt the approach proposed in [11]. This approach can be explained as follows. For each pixel $\boldsymbol{X}_n \in \mathcal{X}$ (we don't consider the boundary pixels which number is negligible as compared to the whole image pixels), there is an immediate neighbor $\widehat{\boldsymbol{X}}_n \in \mathcal{X}$ which is supposed to have arisen from the same cluster of $\boldsymbol{X}_n$, we call it the peer of $\boldsymbol{X}_n$. Since it is supposed that the peers stay in the same clusters, this spatial information can be used as indirect information for estimating the number of clusters. In this scenario, if a larger value is assigned to M, there would be a conflict with the indirect information, provided by the pixels spatial repartition, of M, which means that a true cluster is wrongly divided into two sub-clusters. These two sub-clusters have then to be merged to form a new cluster which related parameters have to be estimated again. In this case, one of the clusters' mixing probabilities will drop suddenly and approaches zero, that can be neglected easily, so the number of clusters will gradually decrease to reach the true number of clusters (i.e. image regions).

Let $\mathcal{X}$ and the set of peers $\widehat{\mathcal{X}} = \{\widehat{\boldsymbol{X}}_1, \ldots, \widehat{\boldsymbol{X}}_N\}$ be our observed data. The set of group indicators for all pixels $\mathcal{Z} = \{\boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_N\}$ will form the unobserved

data, where $\mathbf{Z}_n = (z_{n,1}, \ldots, z_{n,M})$ denotes the missing group indicator and $z_{n,j}$ is equal to one if $\mathbf{X}_n$ and $\widehat{\mathbf{X}}_n$ belong to the same cluster j, or zero, otherwise. The complete data log-likelihood is given by

$$L(\mathcal{X}, \widehat{\mathcal{X}}, \mathcal{Z}|\Theta) = \sum_{n=1}^{N} \sum_{j=1}^{M} z_{n,j}(2 \log P_j + \log p(\mathbf{X}_n|\boldsymbol{\theta}_j) + \log p(\widehat{\mathbf{X}}_n|\boldsymbol{\theta}_j)) \quad (1)$$

An important property of the GD mixture, previously shown in [5], is that the problem of estimating its parameters can be reduced to estimation of the parameters of d Beta mixtures by using the following transformation: $T(X_{nd}) = X_{nd}$ if $d = 1$ and is equal to $\frac{X_{nd}}{1 - X_{n1} - \ldots - X_{nd-1}}$, otherwise. Indeed, in the constructed vector $\mathbf{W}_n = (W_{n1}, \ldots, W_{nD})$, each W_{nd}, $d = 1, \ldots, D$, has a Beta distribution with parameters α_{nd} and β_{nd} [5]. According to this property, the estimation problem is reduced to the optimization of

$$L(\mathcal{W}, \widehat{\mathcal{W}}, \mathcal{Z}|\Theta_d) = \sum_{n=1}^{N} \sum_{j=1}^{M} z_{n,j}(2 \log P_j + \log p_{\text{beta}}(\mathbf{W}_{nd}|\boldsymbol{\theta}_{jd}) + \log p_{\text{beta}}(\widehat{W}_{nd}|\boldsymbol{\theta}_{jd}))$$

$$(2)$$

where p_{beta} denotes the Beta distribution, $\mathcal{W} = (W_{1d}, \ldots, W_{Nd})$, $0 < d < D$, $\boldsymbol{\theta}_{jd} = (\alpha_{jd}, \beta_{jd})$ and $\Theta_d = (\{P_j\}, \{\boldsymbol{\theta}_{jd}\})$. The most popular algorithm to estimate the parameters of a finite mixture model is the EM algorithm which can be done iteratively via 2 different steps: the expectation (E) step and maximization (M) step. In E-step, the conditional expectation of $L(\mathcal{W}, \widehat{\mathcal{W}}, \mathcal{Z}|\Theta_d)$ is calculated:

$$Q(\mathcal{W}, \widehat{\mathcal{W}}, \Theta_d) = \sum_{n=1}^{N} \sum_{j=1}^{M} p_{\text{beta}}(j|\mathbf{W}_{nd}, \widehat{\mathbf{W}}_{nd}, \boldsymbol{\theta}_{jd}) \quad (3)$$

$$\times (2 \log P_j + \log p_{\text{beta}}(\mathbf{W}_{nd}|\boldsymbol{\theta}_{jd}) + \log p_{\text{beta}}(\widehat{\mathbf{W}}_{nd}|\boldsymbol{\theta}_{jd}))$$

where $p_{\text{beta}}(j|\mathbf{W}_{nd}, \widehat{\mathbf{W}}_{nd}, \boldsymbol{\theta}_{jd})$ is the posterior probability that $\mathbf{W}_{nd}$ and $\widehat{\mathbf{W}}_{nd}$ are assigned to cluster j:

$$p_{\text{beta}}(j|\mathbf{W}_{nd}, \widehat{\mathbf{W}}_{nd}, \boldsymbol{\theta}_{jd}) = \frac{P_j p_{\text{beta}}(\mathbf{W}_{nd}|\boldsymbol{\theta}_{jd}) P_j p_{\text{beta}}(\widehat{\mathbf{W}}_{nd}|\boldsymbol{\theta}_{jd})}{\sum_{j'=1}^{M} P_{j'} p_{\text{beta}}(\mathbf{W}_{nd}|\boldsymbol{\theta}_{j'd}) P_{j'} p_{\text{beta}}(\widehat{\mathbf{W}}_{nd}|\boldsymbol{\theta}_{j'd})} \quad (4)$$

Then, in M-step, $Q(\mathcal{W}, \widehat{\mathcal{W}}, \Theta_d)$ is maximized which gives us the following for P_j:

$$P_j^{(k+1)} = \frac{1}{N} \sum_{n=1}^{N} p_{\text{beta}}(j|\mathbf{W}_{nd}, \widehat{\mathbf{W}}_{nd}, \theta_{jd}^{(k)}) \quad (5)$$

As for the θ_{jd} parameters, we shall employ a Newton-Raphson approach:

$$\boldsymbol{\theta}_{jd}^{(k+1)} = \boldsymbol{\theta}_{jd}^{(k)} - H^{-1}(\boldsymbol{\theta}_{jd}^{(k)}) \times \left(\frac{\partial Q(\mathcal{W}, \widehat{\mathcal{W}}, \theta)}{\partial \boldsymbol{\theta}_{jd}}\right) \quad (6)$$

where H is the Hessian matrix, which requires the calculation of the second and mixed derivatives of $Q(\mathcal{W}, \widehat{\mathcal{W}}, \Theta_d)$ which can be straightforwardly done in the same way as in [5]. Having the required estimation equations in hand, the proposed segmentation algorithm can be summarized as follows:

1. Choose a large initial value for M as number of image regions (this value should be larger than the expected number of the regions in the image).
2. Initialize the algorithm using the approach in [5].
3. Use the image data points and related peer points to update generalized Dirichlet mixture parameters by alternating the following two steps:
   - E-Step: Compute the posterior probabilities using equation 4.
   - M-Step: Update the mixture parameters using equations 5 and 6.
4. Check the P_j values. If a value is close to 0 its related cluster should be removed and the number of clusters, M, should be reduced by 1.
5. Go to 3 until convergence.

Since for each pixel (r, c) there are 4 main neighbors that are likely to be in the same region, we can use one of them as the corresponding peer of the pixel. In our experiments, we shall use the pixel $(r + 1, c)$ as the corresponding peer. It is noteworthy that the proposed algorithm is different from classic approaches for mixture learning which are generally composed of two components: a parameters estimation algorithm and a criterion for comparing models where each model is characterized by a certain number of components. In our case here the selection of the number of image regions is done simultaneously with the estimation process.

3   Experimental Results

In this section, we demonstrate the performance of our approach as compared to [11]. We considered also the case where the Dirichlet [12] is considered instead of the GD. We also examine the influence of the color space choice. In particular, we compare two color spaces namely the RGB normalized color space which rgb planes defined by [16]: $r(R, G, B) = \frac{R}{R+G+B}$, $g(R, G, B) = \frac{G}{R+G+B}$, $b(R, G, B) = \frac{B}{R+G+B}$ and the $l_1l_2l_3$ color space defined by [16]: $l_1(R, G, B) = \frac{(R-G)^2}{(R-G)^2+(R-B)^2+(G-B)^2}$, $l_2(R, G, B) = \frac{(R-B)^2}{(R-G)^2+(R-B)^2+(G-B)^2}$, $l_3(R, G, B) = \frac{(G-B)^2}{(R-G)^2+(R-B)^2+(G-B)^2}$. The rgb and $l_1l_2l_3$ have been shown to outperform the widely used RGB space [16]. To have a fair comparison, in all cases (i.e. Gaussian, Dirichlet and GD mixtures), the initial value of M is set to 30.

In addition to the famous Baboon image, which is widely used to evaluate image segmentation algorithms, we have employed our approach on 300 images from the well-known publicly available Berkeley segmentation data set [17]. This database is composed of a variety of natural color images generally used as a reliable way to compare image segmentation algorithms. Figure 1 shows a comparison between the segmentation results obtained by our approach and the technique developed in [11] when we consider the rgb color space. Figure 1(a) shows

the original Baboon image, while figures 1(b), 1(c) and 1(d) show the results obtained with the Gaussian, the Dirichlet and the GD mixtures, respectively. The algorithm in [11] selected 12 regions for the baboon image while 4 regions are considered with both the Dirichlet and GD mixture models. As the figure indicates, in addition to the less number of regions preferred by our algorithm, the regions provided are more meaningful. In this image the nose of baboon is almost composed of two clear regions while the hair is divided to light and dark regions. The images in figure 2 are chosen from the Berkeley database. The estimated

(a)   (b)   (c)   (d)
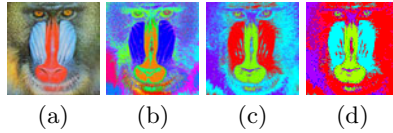
Fig. 1. Baboon segmentation in the rgb space. (a) Original image, (b) Using the Gaussian mixture ($M = 12$), (c) Dirichlet mixture ($M = 4$), (d) GD mixture ($M = 4$).

number of regions, in these images when considering the rgb color space, selected by the Gaussian, the Dirichlet and the GD mixtures are mentioned in the related images sub-captions. According to this figure we can see clearly that Dirichlet mixture-based segmentation algorithms generate both quantitatively (less and better regions) and qualitatively (more meaningful) better results when compared to the Gaussian mixture. The GD outperforms generally the Dirichlet as shown, for instance, in figures 2(g) 2 (h) which is actually expected since the Dirichlet is just a special case of the GD when $\beta_d = \alpha_{d+1} + \beta_{d+1}$. Evaluating

(a)   (b) $M = 6$ (c) $M = 3$ (d) $M = 3$

(e)   (f) $M = 8$ (g) $M = 4$ (h) $M = 4$

(i)   (j) $M = 4$ (k) $M = 3$ (l) $M = 3$

(m)   (n) $M = 7$ (o) $M = 3$ (p) $M = 3$
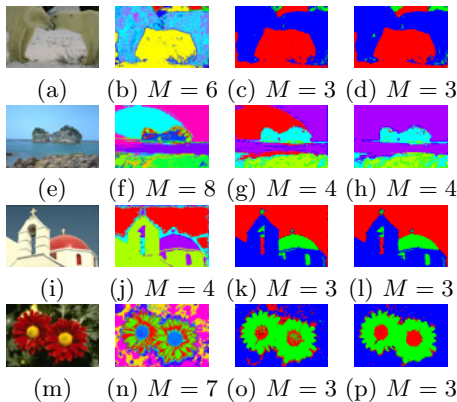
Fig. 2. Examples of images segmentation results in the rgb color space. (a,e,i,m) Original images from the Berkeley Database. (b,f,j,n) Segmentation results using the Gaussian mixture. (c,g,k,o) Dirichlet mixture. (d,h,l,p) Generalized Dirichlet mixture.

segmentation results is an important problem and several quantitative evaluation measures have been proposed in the past (see, for instance, [18,19,20,21]. In our case we have used the Normalized Probabilistic Rand (NPR) index, proposed in [21], as a neutral scale for quantitative comparison between image segmentation algorithms. The NPR index has a value between -1 to 1 where a higher value shows better segmentation results. The calculation of the NPR index requests the availability of a hand-labeled segmentation used as a ground truth to score the segmentation algorithm. The Berkeley database has provided at least 5 ground truth segmentation results for all its 300 natural public images. Because of the "expensive" calculation of NPR Index [21], we have calculated the NPR index for a reasonable number of images. The results indicate that the Dirichlet mixture has a sample mean of 0.5376 for NPR index for some randomly selected images, the GD algorithm has a sample of 0.5523 for the same images, while the Gaussian mixture-based algorithm has a sample mean equal to 0.2667. In all cases we have considered the rgb color space. Figure 3 shows some of the original images used to calculate the NPR index, the Gaussian mixture's segmentation results, and our segmentation results. It shows also five ground truth segmentations, from Berkeley database, for each selected image. The NPR index for each algorithm is mentioned in the images sub-captions. Figure 4 illustrates the effect

NPR=0.4511 NPR=0.6722 NPR=0.6761

NPR=0.2978 NPR=0.3213 NPR=0.4913

NPR=0.3996 NPR=0.5712 NPR=0.5744

NPR=0.2871 NPR=0.5411 NPR=0.5588
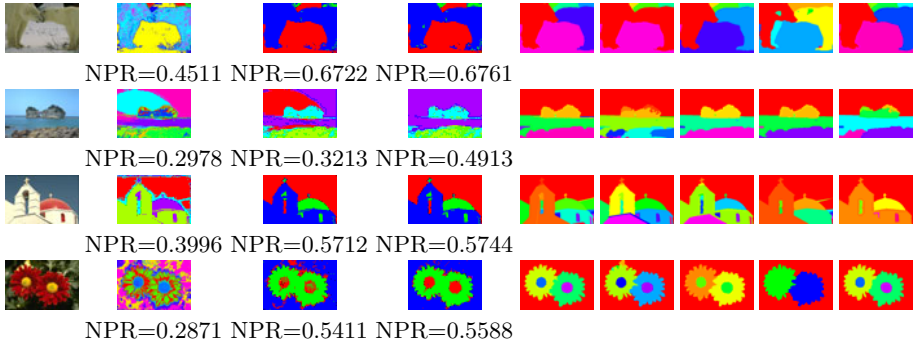
Fig. 3. Examples of images used to calculate the NPR index of each segmentation approach in the rgb color space. Column 1: the original images. Column 2: the segmentation results using the Gaussian mixture. Column 3: Dirichlet mixture. Column 4: Generalized Dirichlet model. Columns 5, 6, 7, 8 and 9: Ground truth segmentations.

of choosing the $l_1l_2l_3$ color space on the segmentation of the Baboon image. According to this image, the new color space has improved the segmentation result in the case of the Gaussian mixture by decreasing the number of regions to 10 as compared to the 12 regions found when the rgb color space has been considered. Changing the color space has provided smoother results in the case of the Dirichlet and GD mixtures while keeping the same number of regions. Figure 5 displays the segmentation results in the $l_1l_2l_3$ color space when considering the images from the Berkeley database. The NPR index values are mentioned in the
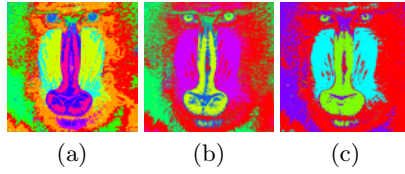
Fig. 4. Baboon segmentation in the $l_1l_2l_3$ space. (a) Using the Gaussian mixture ($M = 10$), (b) Dirichlet mixture ($M = 4$). (c) Generalized Dirichlet mixture ($M = 4$).

sub-captions. As we can see from this figure, choosing the $l_1l_2l_3$ space provides generally smoother and more meaningful regions.

NPR=0.4777   NPR=0.6783   NPR=0.6822

NPR=0.3101   NPR=0.3556   NPR=0.5081

NPR= 0.4015   NPR=0.5817   NPR=0.5823

NPR= 0.3092   NPR=0.5553   NPR=0.5595
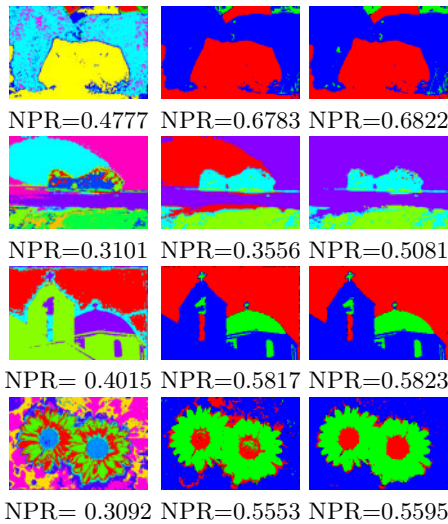
Fig. 5. Segmentation in the $l_1l_2l_3$ color space. column 1: Using the Gaussian mixture model. Column 2: Dirichlet mixture. Column 3: Generalized Dirichlet mixture.

4   Conclusion

An important problem in computer vision is that of segmenting images into homogeneous regions. We tackle this problem, in this paper, by using maximum likelihood estimation of generalized Dirichlet mixture models augmented with spatial information via the incorporation of a physical constraint of the segmentation problem namely neighboring pixels are supposed to belong to the same region. The proposed method is illustrated by the segmentation of several challenging images and is shown to achieve significant performance gains over previously proposed comparable approaches.

References

1. Vasconcelos, N., Kunt, M.: Content-Based Retrieval from Image Databases: Current Solutions and Future Directions. In: Proc. of ICIP, pp. 6–9 (2001)
2. Yu, S.X., Shi, J.: Segmentation Given Partial Grouping Constraints. IEEE TPAMI 26(2), 173–183 (2004)
3. Allili, M.S., Bouguila, N., Ziou, D.: Finite General Gaussian Mixture Modeling and Application to Image and Video Foreground Segmentation. Journal of Electronic Imaging 17(1), 1–13 (2008)
4. McLachlan, G.J., Peel, D.: Finite Mixture Models. Wiley, New York (2000)
5. Bouguila, N., Ziou, D.: A Hybrid SEM Algorithm for High-Dimensional Unsupervised Learning Using a Finite Generalized Dirichlet Mixture. IEEE Trans. on Image Processing 15(9), 2657–2668 (2006)
6. Ton, J., Sticklen, J., Jain, A.K.: Knowledge-Based Segmentation of Landsat Images. IEEE Trans. on Geoscience and Remote Sensing 29(2), 222–232 (1991)
7. Ng, S.-K., McLachlan, G.J.: Speeding up the EM Algorithm for Mixture Model-Based Segmentation of Magnetic Resonance Images. Pattern Recognition 37, 1573–1589 (2004)
8. Mirmehdi, M., Petrou, M.: Segmentation of Color Textures. IEEE TPAMI 22(2), 142–159 (2000)
9. Montanvert, A., Meer, P., Rosenfeld, A.: Hierarchical Image Analysis Using Irregular Tessellations. IEEE TPAMI 13(4), 307–316 (1991)
10. Liu, J., Yang, Y.-H.: Multiresolution Color Image Segmentation. IEEE TPAMI 16(7), 689–700 (1994)
11. Yang, X., Krishnan, S.M.: Image Segmentation using Finite Mixtures and Spatial Information. Image and Vision Computing 22(9), 735–745 (2004)
12. Bouguila, N., Ziou, D., Vaillancourt, J.: Unsupervised Learning of a Finite Mixture Model Based on the Dirichlet Distribution and its Application. IEEE Trans. on Image Processing 13(11), 1533–1543 (2004)
13. Boutemedjet, S., Ziou, D., Bouguila, N.: Unsupervised Feature Selection for Accurate Recommendation of High-Dimensional Image Data. In: Advances in Neural Information Processing Systems (NIPS), pp. 177–184 (2007)
14. Bouguila, N., Ziou, D.: Dirichlet-Based Probability Model Applied to Human Skin Detection. In: Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 521–524 (2004)
15. Bouguila, N., Ziou, D.: A Powerful Finite Mixture Model Based on the Generalized Dirichlet Distribution: Unsupervised Learning and Applications. In: Proc. of the 17th International Conference on Pattern Recognition (ICPR), pp. 280–283 (2004)
16. Gevers, T., Smeulders, A.W.M.: Color-Based Object Recognition. Pattern Recognition 32(3), 453–464 (1999)
17. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics. In: ICCV, pp. 416–423 (2001)
18. Borsotti, M., Campadelli, P., Schettini, R.: Quantitative Evaluation of Color Image Segmentation Results. Pattern Recognition Letters 19, 741–747 (1998)
19. Chang, K.I., Bowyer, K.W., Sivagurunath, M.: Evaluation of Texture Segmentation Algorithms. In: Proc. of CVPR, pp. 294–299 (1999)
20. Everingham, M., Muller, H., Thomas, B.: Evaluating Image Segmentation Algorithms using the Pareto Front. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2353, pp. 34–48. Springer, Heidelberg (2002)
21. Unnikrishnan, R., Pantofaru, C., Hebert, M.: Toward Objective Evaluation of Image Segmentation Algorithms. IEEE TPAMI 29, 929–944 (2007)

A Motion Detection Model Inspired by Hippocampal Function and Its FPGA Implementation

Haichao Liang and Takashi Morie

Graduate School of Life Science and Systems Engineering,
Kyushu Institute of Technology, Kitakyushu 808-0196, Japan

Abstract. We propose a motion detection model inspired by hippocampal function and its FPGA implementation. The proposed model detects the motion of edges extracted from monocular image sequences. The motion is detected on segmented 2D maps without image matching, which allows the model to operate with higher speed than the video rate. We introduce gating units into our original CA3-CA1 model to improve the detection rate, where CA3 and CA1 are the names of hippocampal regions. We have evaluated the performance of our model by using artificial and real image sequences. The results show that the proposed model can achieve high detection rate. We have implemented the model into an FPGA, by which we can achieve motion detection within 1.0 msec/frame with power dissipation of about 1.4 W when 64×60 segmented blocks are used for 320×240 pixel images.

Keywords: edge-based motion detection, time-to-travel, approaching object detection, FPGA implementation.

1 Introduction

Motion detection is one of the key issues in collision warning systems for vehicles or mobile robots. There have been many advances that work by analyzing two-dimensional image motion, which is known as optical flow field [2,7]. Among these, gradient models and block matching models have been popular and have been implemented in OpenCV [9]. However, these still have problems to improve the detection accuracy as well as to reduce the computational cost.

In our research, we aim to propose a collision warning system that can run in high speed and low computational cost. We have proposed such a system inspired by the neuronal propagation in the hippocampus in the brain [5]. The system treats edges extracted from monocular image sequences, and detects motion of the edges without image matching, by using a so-called CA3-CA1 model [12,3]. Here, CA3 and CA1 are the names of hippocampal regions. We used this CA3-CA1 model to detect moving edges as a spatiotemporal pattern, which is essential in the fuzzy-based danger evaluation in our system [5].

The motion detection algorithm of the CA3-CA1 model is similar to those of time-to-travel models which treat particular image features and measure the
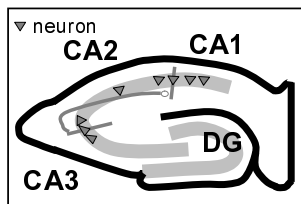
Fig. 1. Hippocampal formation

time that a feature takes to travel between two fixed locations [4,8]. The measured time is called *travel time* in this paper. Time-to-travel models were mainly proposed for applications in some simple situations, e.g. the camera is stationary or there is few moving objects in a simple background.

In this paper, we introduce gating units into the original CA3-CA1 model, and use the function of gating units to reduce mismatching and improve the detection rate in complicated situations. Then, we propose an FPGA implementation of the proposed model for high-speed and low cost processing. We evaluate the performance of the proposed model by using artificial and real image sequences.

2   Proposed Model for Motion Detection

A basic motion-detection model named CA3-CA1 model [12,3] has been proposed inspired by the neuronal propagation in the hippocampus. The hippocampus is known as an area of the brain related with memory function. The hippocampal formation consists of two principal regions: the dentate gyrus (DG), and the cornu ammonis (CA), where CA are usually divided into CA1, CA2 and CA3 by the anatomical difference, as shown in Fig. 1. It is verified that the CA3-CA1 model can successfully detect motion of edges and is useful for collision warning in simple situations [5]. However, the CA3-CA1 model has a trouble with motion detection in complicated situations.

In order to improve the model, we reviewed the neuronal propagation in the hippocampus. According to [1,6], CA2 neurons receive inputs in parallel with DG, and send inhibitory signals to CA1 neurons. Another research [10,11] indicated that the propagation between CA3 and CA1 is in two pathways: one is fast propagation in a CA3-CA1 pathway, the other is slow propagation in a CA3-CA2-CA1 pathway, where CA2 neurons can function as a gate. We took some hints from these knowledges to improve the original model.

The proposed model is shown in Fig. 2. In this model, four kinds of 2D maps are employed: Actual image (AC) map, CA3 map, CA1 map and GU map. The AC and CA3 maps are divided into two submaps with different map-divisions, and each CA3 submap is related to two corresponding CA1 submaps that detect motion of edges in opposite directions in horizontal and vertical directions, respectively. These maps were divided into pieces based on a specified method, where a piece of them is called a *unit*.
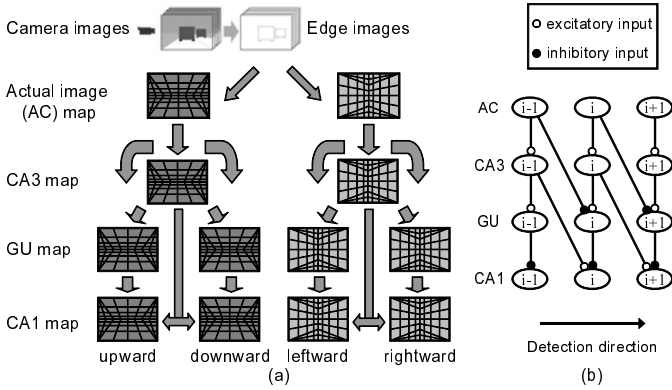
Fig. 2. Proposed motion detection model: (a) model structure and (b) unit connection
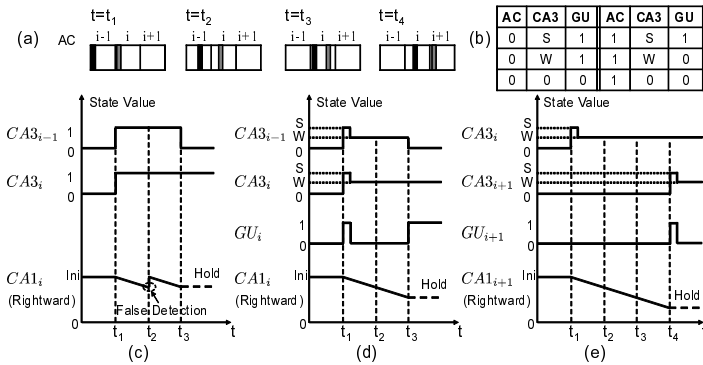
Fig. 3. Motion detection using the proposed algorithm: (a) inputs of AC units, (b) truth table to determine GU value, (c) operation in original algorithm, (d) and (e) operation in proposed algorithm

We introduce gating units (GU) into the model, and assume that each GU receives an excitatory input from the corresponding CA3 unit and an inhibitory input from the backward neighboring AC unit, which means that GU_i receives an inhibitory input from AC_{i-1}. Each GU sends an inhibitory input to the corresponding CA1 unit.

The proposed algorithm is shown in Fig. 3. The value of GU is binary. CA3 has either a strong firing value (S) higher than the GU value or a weak firing value (W) lower than that. At the first firing time, a CA3 unit is set to the strong value, and this value will decay to the weak value before the next sequence is input. After a CA3 unit receives a firing signal, it immediately fires and sets the neighboring CA1 units to decay from the initial value to zero. GU functions as a gate in the motion detection procedure. If a GU is inhibited, its corresponding

CA1 unit can remain decaying, otherwise the CA1 unit turns to be inhibited and stops decaying to hold its value for motion detection.

Therefore, when a CA3 unit fires for the first time, a motion vector is detected irrespective of the corresponding GU state, if one of its corresponding CA1 units is in the decaying state. In contrast, when a CA3 unit fires after the first time, the CA3-GU-CA1 pathway is shut off if the corresponding GU is inhibited, and no motion is detected in this pathway.

In our model, the velocity of a moving edge is calculated by the detected travel time, as shown in Eq. (1), because the distance (D) between two neighboring units is predefined. Therefore, it is important to detect the travel time accurately.

$$v = \frac{D}{\text{travel time}} = \frac{D}{(\text{CA1 Initial Value} - \text{CA1 Value})/\text{Decay coefficient}} \quad (1)$$

Fig. 3 shows an example of motion detection in a complicated situation. In applications to collision warning systems for vehicles or mobile robots, because the background and moving objects produce different optical flows in the image, there must be a moment at which two neighboring CA3 units are in the firing state simultaneously. The motion detection using original CA3-CA1 model has a trouble in such situation, as shown in Fig. 3(c). This problem becomes critical when the background is complicated or a few moving objects exist in the image.

In the example shown in Fig. 3, an edge is moving from AC_{i-1} to AC_i while another edge is moving into AC_i. If using original CA3-CA1 model, false motion is detected at $t = t_2$ when these two edges slightly moved inside of AC_{i-1} and AC_i, respectively. Furthermore, because $CA1_i$ is reset to the initial value after the detection at $t = t_2$, the actual travel time ($t_3 - t_1$) can not be correctly detected at $t = t_3$, as shown in Fig. 3.

In contrast, the proposed model can successfully detect motion at $t = t_3$ without an error at $t = t_2$, as shown in Fig. 3(d). Although GU_i fires at $t = t_1$, this has no effect on motion detection in discrete-time calculation, because $CA1_i$ is still in the initial state. Fig. 3(e) shows motion detection by using strong firing of CA3 units. In this case, although GU_{i+1} has an inhibitory input from AC_i, a motion vector is detected by strong firing of $CA3_{i+1}$.

As a summary, the proposed model detects motion by using not only the corresponding AC unit but also the neighboring AC units to stop the decay in the CA1 unit. Therefore, the proposed model can detect motion more robustly than the original one, so that it can reduce the false detection rate.

3   FPGA Implementation

We have implemented the proposed model in an FPGA in order to develop a high-speed and low-cost motion detection system. We have implemented the CA3-GU-CA1 network in an FPGA, and implemented the AC map in software because the AC map has a large amount of input data and low computation complexity.
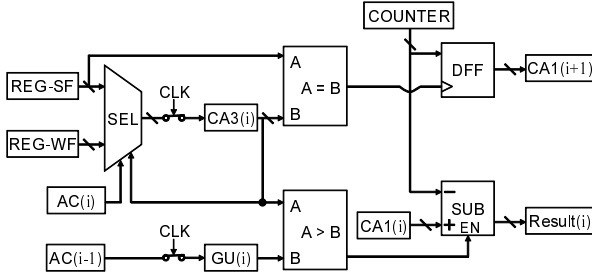
Fig. 4. Digital circuit for FPGA implementation

In our FPGA architecture, we composed CA3, GU and CA1 maps with embedded memory blocks because the utilization of memory blocks can significantly reduce the complexity of digital circuits, so that we can reduce the utilization of LUTs in the FPGA. We designed a row-by-row parallel processing circuit for each CA3/GU/CA1 submap by using the embedded memory blocks.

The proposed digital circuit for FPGA implementation is shown in Fig. 4. We used a common counter for the decaying signal generator to measure travel time. *REG-SF* and *REG-WF* in Fig. 4 indicate the strong and weak firing value of CA3 units, respectively. The specifications of the implemented model and the implementation results are shown in Table 1 and Table 2, respectively.

The maximum frequency of operation is 120.57 MHz, and the processing time is less than one msec/frame when the FPGA operates at 48 MHz. The power dissipation is about 1.4 W according to the power analysis by the EDA software of Altera Quartus®II 9.0.

Table 1. Model Specifications

Image size	320 × 240 pixels (W × H)	
Number of submaps	CA3 : 2, GU : 4, CA1 : 4	
Number of units	64 × 60 units/submap	
CA1 decay steps	64 steps	
Sampling frequency	30 fps	240 fps
Detection range	2–120 km/h	15–960 km/h

4   Performance Evaluation

Let us denote the number of units in the cases shown in Table 3 by TP, FP, FN and TN, respectively. The evaluation indicators of our simulation were defined as follows:

$$\text{Precision Rate} = \frac{TP}{TP + FP}, \ \text{Recall Rate} = \frac{TP}{TP + FN}. \tag{2}$$

Table 2. Device Utilization for Altera EP2S60F672C

Resource	Used	Avail	Utilization
LUTs	13201	48352	27.30 %
Registers	5484	48352	11.34 %
Memory Bits	270336	2544192	10.62 %

Table 3. Denotations of Evaluation Indicators

Results	Edges moved to a neighboring unit	
	Yes	No
Motion Detected	True Positive (TP)	False Positive (FP)
No Motion Detected	False Negative (FN)	True Negative (TN)

These two evaluation indicators are important to our model because our model detects motion only when an edge moved from a unit to its neighboring one using sparse data (edge image). *Precision rate* indicates how well the detected motion can be trusted, and *Recall rate* indicates how well our model detects occurred motion.

We first evaluated the performance of our proposed model by using artificial image sequences, because it is difficult to obtain such real image sequences that contain ground truth of optical flows of the same scenes in different frame rates. We used a picture shown in Fig. 5(a) as a stationary background, and used two cubes shown in Fig. 5(b) as a receding object RO and an approaching object AO moving at a speed of $v_Z = \pm 80 km/h$, respectively.

The simulation results are shown in Table 4. We verified that the recall rate of the proposed model reaches about 99 %, which is much better than the original model. The precision rate was about 81 % at 30 fps, and it was achieved to about 86 % at 240 fps. Furthermore, using a high frame rate can reduce quantization errors in time, which contributes to accurate travel time detection and velocity calculation according to Eq. (1).

Simulation results using real image sequences are shown in Fig. 6. In this simulation, we saved motion detection results for five frames, and used a
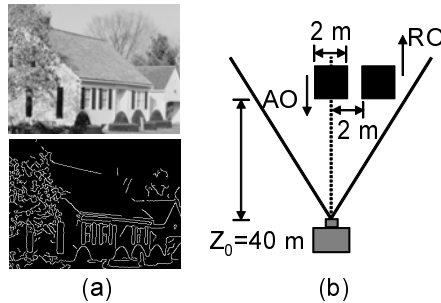
Fig. 5. Simulation patterns used in the artificial image sequences: (a) a picture and its edge image used as a stationary background, (b) patterns of two moving objects

Table 4. Evaluation Results Using Artificial Image Sequences

Model	Frame Rate	Precision	Recall
Original	30 fps	85.5 %	75.4 %
	240 fps	86.9 %	78.0 %
Proposed	30 fps	80.9 %	99.1 %
	240 fps	86.0 %	98.4 %
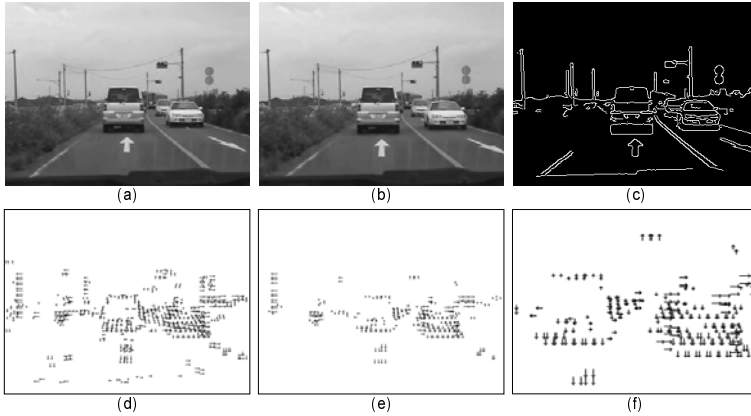
Fig. 6. Simulation results using real image sequences: (a), (b) input images of frames #39 and #43, (c) edge image of frame #43 by using Canny edge detection, (d) superimposed results of motion detection for consecutive five frames, (e), (f) results after simple post-processing, and (f) magnified view of the vehicle regions

morphological filter to delete isolated results in each frame. By Fig. 6(e) and (f), we verified that motion of approaching objects is successfully detected, and false positive results can be partly deleted by the simple morphological filter. This post-processing can be improved by using motion analysis. Therefore, the proposed model can be applied to high-speed detection in complicated situations after we improved image feature detection and the post-processing.

5   Conclusion

In this paper, we proposed a motion detection model and its FPGA implementation inspired by the neuronal propagation in the hippocampus in the brain. We introduced gating units into the original CA3-CA1 model to improve the detection rate (recall rate), and implemented the proposed model in an FPGA. Performance evaluation results using artificial image sequences showed that the detection rate of moving edges was significantly improved. We also presented simulation results using real image sequences with a simple morphological filter to filter out the origins of false positive results in the post-processing. In future work, we will improve image feature detection and the post-processing.

References

1. Bartesaghi, R., Migliore, M., Gessi, T.: Input-output relations in the entorhinal cortex-dentate-hippocampal system: Evidence for a non-linear transfer of signals. Neuroscience 142(1), 247–265 (2006)
2. Beauchemin, S.S., Barron, J.L.: The computation of optical flow. ACM Computing Surveys 27(3), 433–467 (1995)
3. Hayashi, H., Nakada, K., Morie, T.: Moving object detection algorithm inspired by the sequence detection in the hippocampus and its digital lsi implementation. In: Brain-Inspired IT III. International Congress Series, vol. 1301, pp. 35–38. Elsevier (2007)
4. Kramer, J., Sarpeshkar, R., Koch, C.: Pulse-based analog VLSI velocity sensors. IEEE Trans. on Circuits and Systems II 44(2), 86–101 (1997)
5. Liang, H., Morie, T., Suzuki, Y., Nakada, K., Miki, T., Hayashi, H.: An FPGA-based collision warning system using hybrid approach. In: 7th Int. Conf. on Hybrid Intelligent Systems (HIS 2007), Kaiserslautern, Germany, pp. 30–35 (2007)
6. Mercer, A., Trigg, H.L., Thomson, A.M.: Characterization of neurons in the CA2 subfield of the adult rat hippocampus. The Journal of Neuroscience 27(27), 7329–7338 (2007)
7. Mitiche, A., Bouthemy, P.: Computation and analysis of image motion: A synopsis of current problems and methods. Int. Journal of Computer Vision 19(1), 29–55 (1996)
8. Nishio, K., Yonezu, H., Furukawa, Y.: A two-dimensional network of analog circuits for motion detection based on the frog visual system. IEICE Trans. on Fundamentals E89-A(2), 428–438 (2006)
9. OpenCV Online Documentation: Video analysis - motion analysis and object tracking (2011), http://opencv.willowgarage.com/
10. Sekino, Y., Obata, K., Tanifuji, M., Mizuno, M., Murayama, J.: Delayed signal propagation via CA2 in rat hippocampal slices revealed by optical recording. Journal of Neurophysiology 78(3), 1662–1668 (1997)
11. Sekino, Y., Shirao, T.: A Role for Signal Propagation Through the Hippocampal CA2 Field in Memory Formation. In: Zhong, N., Liu, J., Yao, Y., Wu, J., Lu, S., Li, K. (eds.) Web Intelligence Meets Brain Informatics. LNCS (LNAI), vol. 4845, pp. 254–266. Springer, Heidelberg (2007)
12. Yoshida, M., Hayashi, H.: Organization of cell assemblies that code temporal sequences in a hippocampal CA3-CA1 model. In: Int. Joint Conf. on Neural Networks (IJCNN 2004), Budapest, Hungary, pp. 495–500 (2004)

An Automated System for the Analysis of the Status of Road Safety Using Neural Networks

Brijesh Verma[1] and David Stockwell[2]

[1] CQUniversity, Queensland, Australia
[2] Transport and Main Roads, Queensland, Australia
{b.verma@cqu.edu.au}

Abstract. This paper presents a neural network based novel automated system that can analyze vehicle mounted video data for improving road safety. There are video data collection systems currently available although no tools exist which could be used to automatically analyze vehicle mounted video data and estimate future crash sites. The main aim of the research presented in this paper is to develop a technique to segment roadside data obtained from vehicle mounted video into regions of interest, classify roadside objects and estimate the risk factor based on roadside conditions and objects for various crashes. A clustering technique for segmentation of roadside frames into regions of interest and a neural network to classify the regions of interest into objects are investigated. The preliminary segmentation and classification results on a small dataset taken from Transport and Main Roads' vehicle mounted video data collection are promising.

Keywords: Neural Networks, Segmentation, Road Safety.

1 Introduction

The Transport and Main Roads (TMR) in Australia collects a variety of data [1-3] to record and ascertain the status of road safety and condition, and uses these data extensively for guiding and justifying road maintenance and capital expenditures. Vehicle mounted video is collected over every state road annually, and has the potential to provide a range of value-added products through advanced image analysis and recognition. There are a range of benefits to TMR from extraction of road defects and risk factors from vehicle mounted video using an automated system. Unfortunately video data is only partially integrated with road management and safety. The potential exists for application of computer processing and analysis of video to automate data collection for improving road safety.

The statistical data relevant to road safety from report [1] produced by Transport and Main Roads show that the 36.3% road fatalities were from hit object type crashes, 17.2% were from head-on type crashes and 14.8% were from angle type crashes. Many of these crashes could be avoided if appropriate sections of roads and risk factors (e.g. objects within clear zone, inappropriate shoulder width, missing sign, etc.) could be identified and fixed. Therefore, collecting information on the road and

roadside is becoming increasingly important for assessments of road safety risk. Currently, collection of information on key risk factors can be a time consuming task and contain some degree of subjective assessment. Collection of data in an automatic way using advanced segmentation, identification and risk factor estimation techniques would improve the degree of accuracy and create safer roads with fewer fatalities.

The traditional road safety approach uses crash history as the basis for assessing risk [2-3]. This type of traditional approach has been highly successful in many countries including Australia but the problem is that many crashes occur at places with no previous crash history. Therefore, recently there has been a move towards an additional risk assessment approach based on the risk inherent in road and roadside features. This type of approach is growing because of the following three reasons: (1) Australia has adopted the safe system approach, intended to understand and address road safety risk by taking a total view of road safety factors that minimize the number of serious injuries and deaths (e.g. address all locations where serious crashes may occur, not just locations where crashes have previously occurred). The components of the safe system are: safer users, safer roads and roadsides, safer vehicles and safer speeds (2) Road authorities must know where risks lie on their roads for legal liabilities. (3) The decreasing number of treatable crash black spots.

Most commercially available systems employing vehicle mounted sensors are intended for surveying and road design, measuring the 3D shape of the road and environment with great precision. However, they are unable to extract road characteristics and objects on roads. Intelligent traffic systems are aimed at easing traffic congestion in urban streets by redirecting the individual motorist into alternative less trafficked routes. Other specialist vehicles employ a range of physical sensors to determine characteristics such as pavement depth. These vehicles are expensive to employ, rarely available, and do not address many of the daily concerns of road maintenance and safety.

In the past few decades, the research with roadside data has been conducted. Although it was mainly focused on road sign detection and classification, number plate recognition, painted road object recognition and road crack identification [4-22]. A number of techniques [4-6, 20-22] for segmentation and extraction of signs has been proposed and investigated. The techniques for extracting symbols, characters and text on road signs [16-17] have also been proposed and investigated. A review of recent literature [4-22] showed that there has been a lot of research with some success in particular road sign recognition, however there has been very little research conducted to automatically extract all types of features and objects from roadside video data for improving road safety and no attempts have been made to develop an automatic data analysis technique to identify sections of roads where a crash may occur. This paper proposes and investigates a new idea of using and analyzing video data for roadside objects and automatically identifying risk factors for all roads and locations where a fatal or serious crash may occur.

This paper is organized as follows. Section 2 presents the proposed approach. Section 3 describes the experimental setup used for evaluating the proposed approach. Section 4 presents some preliminary experimental results and discussion. Finally, Section 5 concludes the paper.

2   Proposed Approach

The proposed approach is based on novel techniques proposed in this paper and our previous research [13-14, 24-30]. The approach consists of three major parts. The first part (Task 1) is to segment road and roadside data into regions of interest such as signs, trees, pedestrian, etc. so this part is focused on development of appropriate segmentation technique. The second part (Task 2) is to identify the segmented regions into one of the objects or features so this part is focused on development of appropriate classification technique for identification of the segmented objects and features. The third and final part (Task 3) is to estimate the risk factor and crash type. An overview of the proposed approach is presented in Fig. 1 below and described in the following sections.
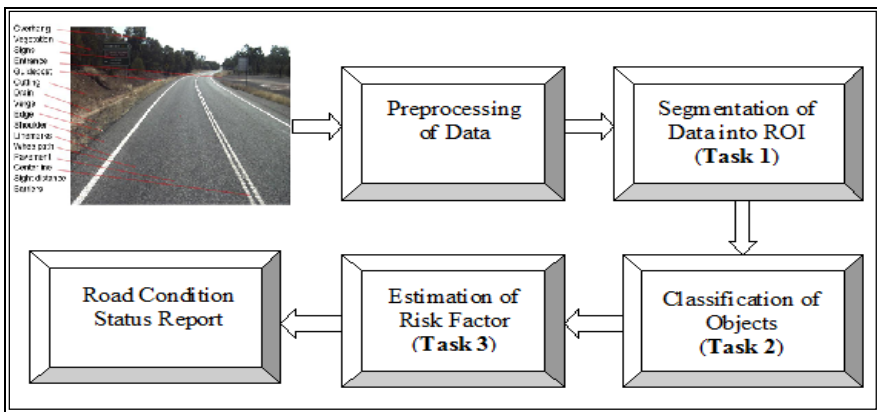
Fig. 1. An overview of the proposed approach

2.1   Pre-processing of Data

Pre-processing involves standard steps for conversion of video data into a series of image data, normalization of colours and other operations to enhance uniformity of image data. A comparative analysis of various colour formats has shown [6, 27, 30] that HSV colour space is most suitable format as it has minimum variance under lighting and other environmental conditions so in this research HSV colour space is used.

2.2   Segmentation of Data into Regions of Interest (ROI)

Segmentation of road and roadside video data is one of the most important steps in identification of various features and hazardous roadside objects. Segmentation step groups the data into following regions of interest (road signs, lane width, shoulder width, clear zone width, road surface condition, separation of opposing traffic flow, delineation, overtaking opportunities/facilities, street lighting, access points, sight distance, etc.). The regions with these features and objects are considered as regions of interest. These regions are considered regions of interest because various studies

[1-3] conducted by ARRB on behalf of Austroads has identified that these roadside features significantly contribute to road safety risk.

Research into segmentation of roadside data has been conducted in the past [3-9, 18-22], however as mentioned in previous sections, the major focus has been on the recognition of traffic signs. Some researchers used segmentation combined with shape classification to detect and classify traffic signs. K-means clustering techniques based on colour feature have been frequently used to segment data into regions not only in traffic sign recognition research but also in other pattern recognition areas such as medical imaging [23, 26, 29]. In this research, we deal with many more objects and features than just traffic signs so the segmentation task is very challenging. The new approach is based on fusion of (1) colour feature (2) texture feature and (3) hierarchical clustering. The fusion of colour and texture is appropriate as we are dealing with road signs and vegetation (road side trees, etc.) for finding various features (e.g. clear zone width, shoulder width, etc.) which have specific colours and textures. The goal of hierarchical clustering is to group data into clusters of similar colour, texture and distance from centre with locally connected pixels. We use top to bottom approach which means finding a large cluster and then dividing it into sub clusters to avoid situations like having many objects (e.g. road signs) in a single cluster. Ideally each cluster represents a single roadside object or feature. We have recently developed a soft clustering based neural network approach and we use this approach from our previous research [13-14, 24-30] to cluster roadside data into regions mentioned above.

Segmentation Approach
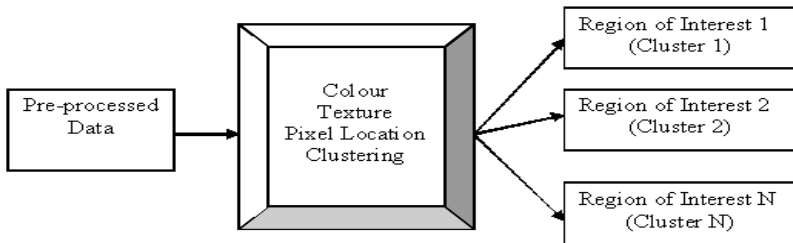
Fig. 2. Segmentation approach

2.3   Classification of Roadside Objects and Features

Most of research on roadside object classification has been focused on sign classification and road cracks. The decision trees [10], likelihood-based classifier [11], and MLP based neural networks [12-13, 18] have been applied. MLP based neural networks achieved over 95% classification accuracy for road signs. Our recent research [24-25] showed that neural network based ensemble can produce very good accuracy. This research have difficult to classify road objects and features under varied conditions, so we focus on evaluating three major classifiers such as SVM, MLP and neural ensemble classifiers which have produced highest accuracy on benchmark datasets in our recent research [24]. A single tier and a two tier classification systems are designed and investigated as shown below.

Single Tier Classification System
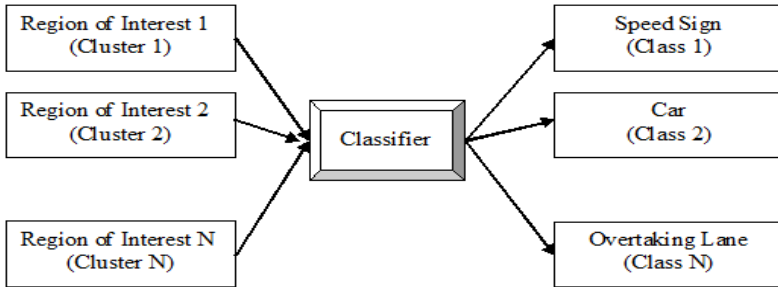
Fig. 3. Single tier classification approach

Two Tier Classification System
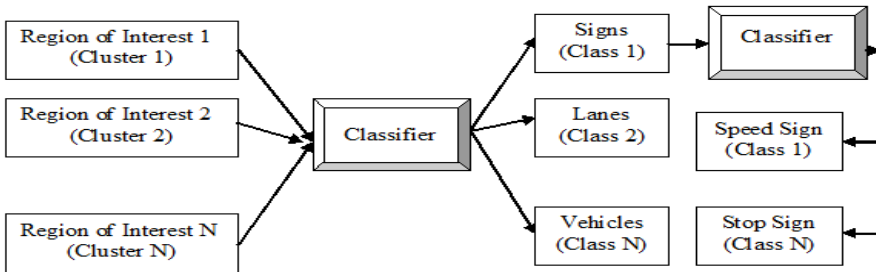
Fig. 4. Two tier classification approach

2.4   Estimation of Risk Factors for All Crash Types

This task focuses on developing a technique to estimate a risk factor for all crash types. Before we start this task, we need to solve two problems as follows. Firstly, we need to know the crash types which we have to deal with and secondly we need to know the features associated with each crash type. The first problem has been solved as various research reports including TMR's report [1-3] and our recent research/discussions have analysed key crash types. It has been identified that the key crash types which are likely to result in high severity outcomes are run-off-road, head-on, intersection, pedestrian and motorcyclist/cyclist. The second problem is very challenging as features for each crash type are not very well defined. We tackle this problem in two ways. The first approach is to investigate and analyse roadside objects and features and define rules for each crash type which will allow us to estimate a risk factor for new data. The second approach is to design a neural network and train it with known crash data. A neural network is chosen because it has shown [24] better generalisation accuracy which might produce better accuracy on unseen crash data.

3 Experiments and Discussion

The database for experiments has been taken from the collection of vehicle mounted video by Transport and Main Roads (TMR) and some other video data taken around our university. Each year, the entire State Road system is recorded with a vehicle mounted video. In addition, a number of specialist vehicles are employed to gather more detailed engineering data, including laser-based imaging and road deflection data. The video data has been converted into frames and then each frame is processed and used for segmentation and classification.

   This research is at its early stage so at the moment we have only conducted some preliminary experiments on segmentation of regions of interest and classification. The segmentation results are shown in Fig. 5 below.
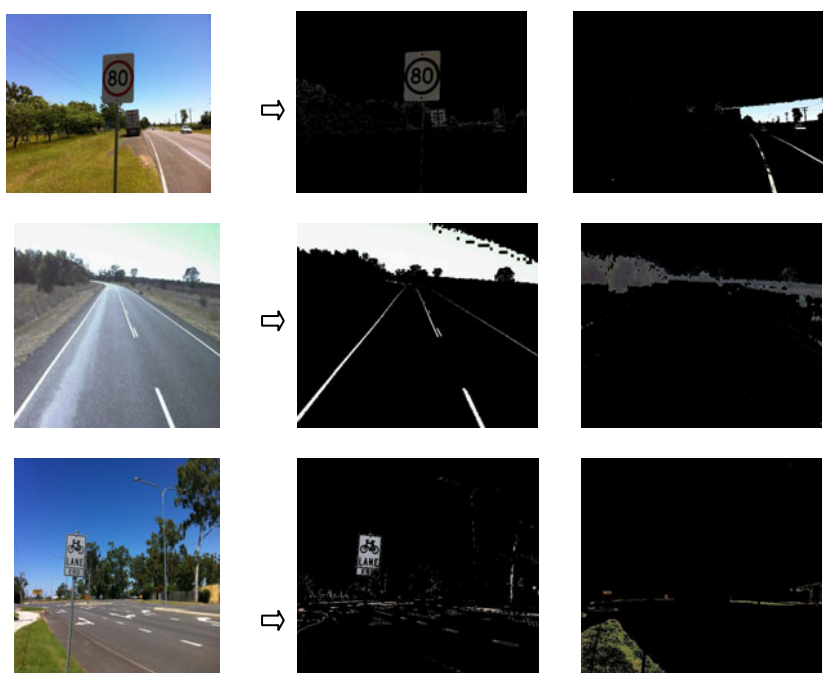
Fig. 5. Segmentation results

   The analysis of segmentation results showed that the proposed approach is able to identify the regions of interest. As shown above, signs, lanes and vegetation have been clearly identified. The regions of interest were trained using neural network for the following categories: vegetation, sky, white [road/road sign background], white [road sign back, road line marker], white [footpath, on road signs], road sign [yellow/yellow vegetation], road sign [red], etc. The classification accuracy on a small number of ROIs mentioned above was very good. All ROIs have been classified correctly but it should be noted that the experiments are very small and more experiments are currently being conducted on a large database taken from TMR.

4   Conclusions

We have proposed a neural network based automated system using vehicle mounted video data for identifying sections of roads where a crash may occur. A technique based on hierarchical clustering for segmentation of roadside data into regions of interest has been investigated. A neural network based classification technique has also been investigated. The preliminary segmentation and classification results are promising and some results are presented in this paper. The results presented in this paper are based on a small database. In future research, we will focus on conducting experiments for segmentation, classification and estimation of risk factor on a large database.

References

1. Road Safety Statistics, `http://www.tmr.qld.gov.au/Safety/Transport-and-road-statistics/Road-safety-statistics.aspx`
2. Turner, B., Tziotis, M., Cairney, P., Jurewicz, C.: Safe System Infrastructure: National Roundtable Report. Research report ARR 370, pp. 1–35 (2009)
3. Turner, B., Comport, L.: Automatic Collection of Safety Related Road and Roadside Data. In: Australasian Road Safety Research Policing Education Conference, pp. 1–10 (2007)
4. Zhang, L., Wu, E.: A Road Segmentation and Road Type Identification Approach Based on New-type Histogram Calculation. In: International Congress on Image and Signal Processing, pp. 1–5 (2009)
5. Xu, G., Xie, S., Yin, Y.: An Optimalizing Threshold Segmentation Algorithm for Road Images Based on Mathematical Morphology. In: International Symposium on Intelligent Information Technology Application, pp. 518–521 (2009)
6. Zakir, U., Leonce, A.N.J., Edirisinghe, E.A.: Road Sign Segmentation Based on Colour Spaces: a Comparative Study. In: International Conference on Computer Graphics and Imaging, pp. 72–79 (2010)
7. Fleyeh, H.: Color Detection and Segmentation for Road and Traffic Signs. In: IEEE Conference on Cybernetics and Intelligent Systems, pp. 809–814 (2004)
8. Kang, Y., Yamaguchi, K., Naito, T., Ninomiya, Y.: Texture Segmentation of Road Environment Scene using SFM Module and HLAC Features. Transactions on Computer Vision and Applications 1, 220–230 (2009)
9. Khan, J., Bhuiyan, S., Adhami, R.: Image Segmentation and Shape Analysis for Road-Sign Detection. IEEE Transactions on Intelligent Transportation Systems 12, 83–96 (2011)
10. Danescu, R., Nedevschi, S.: Detection and Classification of Painted Road Objects for Intersection Assistance Applications. In: 13th IEEE International Annual Conference on Intelligent Transportation Systems, pp. 433–438 (2010)
11. Alvarez, A., López, A.: Road Detection Based on Illuminant Invariance. IEEE Transactions on Intelligent Transportation System 12, 184–193 (2011)
12. Yok-Yen, N., Abbas, K.: Detection and Classification of Road Signs in Natural Environments. Neural Computing and Applications 17, 265–289 (2010)

13. Verma, B., Bray, J., Li, X., He, W.: A Neural Network Based Technique for Automatic Classification of Road Cracks. In: IEEE International Joint Conference on Neural Networks, pp. 1886–1891 (2006)
14. Goyal, A., Verma, B.: A Neural Network Based Approach for the Vehicle Classification. In: IEEE Symposium on Computational Intelligence in Image and Signal Processing, pp. 226–231 (2007)
15. Miura, J., Kanda, T., Shirai, Y.: An Active Vision for Real Time Traffic Recognition. In: IEEE Conference on Intelligent Transportation Systems, pp. 52–57 (2000)
16. Wu, W., Chen, X., Yang, J.: Incremental Detection of Text on Road Signs from Video with Application to a Driving Assistant System. In: Annual ACM International Conference on Multimedia 2004, pp. 852–859 (2004)
17. Wang, K., Kangas, J.A.: Character Location in Scene Images from Digital Camera. Pattern Recognition 36, 2287–2299 (2003)
18. Moreira, J., da Costa, L.F.: Neural-based Color Image Segmentation and Classification Using Self-organizing Maps. In: Anais do IX SIBGRAP, pp. 47–54 (1996)
19. Shinde, G.N., Deshmukh, K.S.: An Adaptive Color Image Segmentation. Electronic Letters on Computer Vision and Image Analysis 5, 12–23 (2005)
20. Shamir, L.: Human Perception-Based Color Segmentation Using Fuzzy Logic. In: International Conference on Image Processing, Computer Vision and Pattern Recognition (IPCV 2006), pp. 496–505 (2006)
21. Zingaretti, P., Tascini, G., Regini, L.: Optimising the Colour Image Segmentation. In: Workshop Sulla Percezione Visione Nelle Machine, pp. 1–6 (2002)
22. Benallal, M., Meunier, J.: Real-time Color Segmentation of Road Signs. In: Canadian Conference on Electrical and Computer Engineering, pp. 1823–1826 (2003)
23. Ng, H.P., Ong, S.H., Foong, K.W.C., Goh, P.S., Nowinski, W.L.: Medical Image Segmentation Using K-Means Clustering and Improved Watershed Algorithm. In: IEEE Southwest Symposium on Image Analysis and Interpretation, pp. 61–65 (2006)
24. Rahman, A., Verma, B.: A Novel Layered Clustering Based Approach for Generating Ensemble of Classifiers. IEEE Transactions on Neural Networks 22, 781–792 (2011)
25. Verma, B., Rahman, A.: Cluster Oriented Ensemble Classifier: Impact of Multi-Cluster Characterisation on Ensemble Classifier Learning. IEEE Transactions on Knowledge and Data Engineering, http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5708148
26. Verma, B., McLeod, P., Klevansky, A.: A Novel Soft Cluster Neural Network for the Classification of Suspicious Areas in Digital Mammograms. Pattern Recognition 42, 1845–1852 (2009)
27. Verma, B., Kulkarni, S.: Neural Networks for Content Based Image Retrieval. In: Semantic-Based Visual Information Retrieval, pp. 252–272 (2006)
28. Verma, B., Muthu, V., Changming, H.: Unsupervised Clustering of Texture Features Using SOM and Fourier Transform. In: IEEE International Joint Conference on Neural Networks, pp. 1237–1242 (2003)
29. McLeod, P., Verma, B.: Clustering and Least Square Based Neural Technique for Learning and Identification of Suspicious Areas within Digital Mammograms. In: International Conference on Computational Intelligence and Multimedia Applications, pp. 190–194 (2007)
30. Verma, B., Kulkarni, S.: A Fuzzy-Neural Approach for Interpretation and Fusion of Colour and Texture Features for CBIR Systems. Applied Soft Computing 5, 119–130 (2004)

Decision Tree Based Recognition of Bangla Text from Outdoor Scene Images

Ranjit Ghoshal[1], Anandarup Roy[2],
Tapan Kumar Bhowmik[3], and Swapan K. Parui[2]

[1] St. Thomas' College of Engineering and Technology
Kolkata- 700023, India
`ranjit.ghoshal@rediffmail.com`
[2] CVPR Unit, Indian Statistical Institute, India
`roy.anandarup@gmail.com, swapan@isical.ac.in`
[3] Faculty of Mathematics and Natural Sciences,
University of Groningen, Netherlands
`tkbhowmik@ai.rug.nl`

Abstract. This article proposes a scheme for automatic recognition of Bangla text extracted from outdoor scene images. For extraction, we obtain the headline, then apply certain conditions to distinguish between text and non-text. By removing the headline we partition the text into two zones. We further observe an association among the text symbols in these two different zones. For recognition purpose, we design a decision tree classifier with Multilayer Perceptron (MLP) at leaf nodes. The root node takes into account all possible text symbols. Further nodes highlight distinguishable features and act as two-class classifiers. Finally, at leaf nodes, a few text symbols remain, that are recognized using MLP classifiers. The association between the two zones makes recognition simpler and efficient. The classifiers are trained using about 7100 samples of 52 classes. Experiments are performed on 250 images (200 scene images and 50 scanned images).

1 Introduction

Automatic recognition of text symbols in a natural scene image is useful to the blind and foreigners with language barrier. Such a recognition method should also employ an extraction of text portions from the scene images. Extraction and recognition of texts from outdoor images captured by such devices is a challenging problem now-a-days due to variations in style, color, background complexity etc. In this article our goal is twofold. One of our preliminary task is to extract the possible text symbols from an input scene image. This has been an issue of interest for many years. Earlier, Jung et. al. [1] employed a multilayer perceptron classifier to discriminate between text and non-text pixels. A good survey of existing methods for detection, localization and extraction of texts embedded in natural scene images can be found in [2]. Considering Bangla script, Bhattacharya et. al. [3] proposed a scheme based on analysis of connected

components for extraction of Devanagari and Bangla texts from camera captured outdoor images. For this purpose the headlines were detected using morphology. We here use this scheme to obtain possible headlines. Afterwards, the text symbols attached/unattached to the headlines are extracted using certain shape and position based purifications.

The recognition issue also draws interest of many scientists. Pal and Chaudhuri [4] present a detailed review on recognition of Indian script. However, all the efforts are still done using scanned documents. To the best of our knowledge no recent works exist that deals with Bangla text separation and recognition from natural scene images.

The headline detection eventually leads us to the zone detection. In Fig. 1 we have shown all the zones.
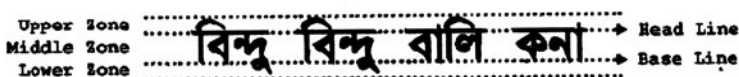
Fig. 1. Illustrate the zones of Bangla Script

In this paper, we merge the middle and the lower zones to form the middle-lower zone. This zone has an association with the upper zone as will be seen later. For recognition in middle-lower zone, we use a decision tree model. At each internal node of this tree, one criterion is evaluated that splits the current set of text symbols into two halves. The leaf nodes, finally, encounter only a few classes and the MLP is used for classification. The association between two zones boost the recognition performance. Our experiments are performed on a laboratory made dataset consisting of both scene and scanned images. The experimental results establish satisfactory performance of our methods.

1.1 Contributions

In this paper we propose a system for extraction and recognition of Bangla text symbols from scene images. The proposed system works in three steps. Initially headlines are detected using mathematical morphology [3]. In the next step, we propose some criteria on the basis of which text symbols are isolated from non-text. These criteria can be applied to scanned images as well, with equal performance, as we show later. The final step of our system is the proposed decision tree classifier. This decision tree automatically recognizes text symbols based on some pre-defined conditions. To boost the recognition performance, we further consider association among text symbols in different zones of a word. These associations lead to small groups of symbols, that are easily recognizable. Such a script based boosting has not yet been done, to the best of our knowledge. Moreover, the proposed system is robust to perspective distortion and image skew at least up to an acceptable level. Also, this system can be extended to scanned documents. In fact, our experiments include scanned documents along with scene images.

2   Headline Based Text Symbol Extraction

In Bangla script, the headline connects most of the text symbols in a word. To detect possible headlines, we apply mathematical morphology based headline detection method reported by Bhattacharya et al [3]. The following subsection describes the procedure to separate text and non text portions.

2.1   Separation of Text Portions

We first proceed with headline attached components. The following criteria are used to identify text portions. All the concerned connected components are subjected to these criteria in the same sequence as given. Note, the thresholds may differ if the sequence is altered.

1. *Boundary attached components*: Generally text like patterns are not attached with boundary of the image. So, we first remove all boundary attached connected components using morphological reconstruction.
2. *Elongatedness ratio* (ER): This was used by Roy et. al. [5]. Empirically it is found that a component with ER value greater than 5 is a text symbol.
3. *Number of complete lobes*: Using Euler number complete lobes can be obtained. Found empirically, a text symbol has less than 9 complete lobes. We simply use lobe, hereafter, since all concerned lobes are complete.
4. *Aspect ratio*: We found by experiments, that the aspect ratio (height/width) of a non-text become less than 0.3 or greater than 2.0.
5. *Object to background pixels ratio* (r): Due to the elongated nature of Bangla script, only a few object pixels fall inside text bounding box. On the other hand, elongated non-texts are usually straight lines, so, contribute enough object pixels. We observe $0.3 \leq r \leq 3$ could identify text symbols.

All the headline attached text symbols are therefore separated.

However, in Bangla, some text symbols do not meet the headline. Now we consider such components. These components, though not connected, must be close to one/more of already detected text symbols. Then, if we increase the area of the bounding box enough, the possible text symbols may lie inside it. With this view, we increase the width of the bounding box by its height and the height by an empirical threshold. The components inside this modified bounding box are subjected to the previous criteria.

2.2   Headline Removal and Zone Detection

We need to remove the headlines to isolate a text symbol. Sometimes a major part of a symbol may be joined with the headline. If the headline is removed totally, these symbols become broken. Instead, we examine all the pixels over the headlines. Let such a pixel p_i (the ith pixel) belongs to a component CC_j. Let v_i be the vertical run length of p_i at the ith position considering only the corresponding component CC_j. We experimentally set the thickness of the

headline to be H (say). Now, if $v_i \leq H$, we can conclude that p_i is a pixel solely from the headline and thus removed.

In previous studies [6], the headline was used to partition a Bangla word into three zones as in Fig. 1. However, to obtain the lower zone, one may need to identify the "*Base Line*" (Fig. 1). This leads to additional tasks. Here, we drop detecting lower zone and proceed with the upper and the middle-lower zones (the latter being the portions below the headline). The symbols in different zones are shown in Fig. 2(a),(b) with class numbers (serial numbers). The numbers in brackets indicate the total number of samples of the corresponding class in our dataset. The three classes: 46, 47 and 48 have lower zones text connected with middle zone text symbols. Let us term these as "composite" symbols. We consider only three lower zone components, ৃ, ু and ৄ . Fig. 2(c) describes how a lower zone symbol is connected with a middle zone symbol.

3   Association between Upper and Middle-Lower Zones

The two zones produced by removing the headline play an important role during recognition. The upper zone contains some specific text symbols. So recognition in the upper zone has a high accuracy. On the other hand, middle-lower zone consists of a large symbol set, so vulnerable for recognition. However, once the upper zone symbol is identified, we could make an assumption about middle zone symbols. Thus, the initial set may be reduced. This assumption is based on prior knowledge about Bangla language. In Bangla, we only have certain pre-defined middle zone symbols for a particular upper zone symbol. Moreover, upper zone symbols exists frequently in Bangla text. So, this very association greatly improves the accuracy of our recognizer. Of course, if all texts are present inside middle-lower zone we could not apply this association. Considering the upper zone text symbols from Fig. 2(b) the corresponding possible middle-lower zone text classes are enlisted in Table 1.

Table 1. Association between upper and middle zone text symbols

Upper zone text class	49	50	51	52
Middle zone text class	2-5, 21, 22, 43, 44	43	43	2, 11, 13, 16, 18, 19, 21-33, 35-38, 40, 41

4   Bangla Text Recognition

The middle-lower zone text symbols may be partitioned into two. One of them consists of all the composite text symbols. The study by Parui et. al. [7] described a method to filter out and recognize lower zone symbols from a composite symbol. After its application, only the middle zone symbols remain. The upper zone symbols are subjected to MLP classifier directly, while a decision tree is applied

Fig. 2. Text components in (a) middle-lower and (b) upper zone. (c) First column shows composite texts and corresponding lower zone symbols are in third column. Serial numbers indicate class numbers.

to the middle zone symbols to form small groups. MLPs are then applied to each such group. For MLP we use the wavelet feature described by Bhowmik et. al. [8].

4.1   Decision Tree Classifier for Middle Zone Symbols

The decision tree evaluates one criterion at each internal node that deterministically splits the current set of text symbols into two subsets. Proceeding this way, each leaf node finally contains a small group of symbols. For each such group, an MLP classifier is used. Before describing the tree, let us concentrate on the leaf level subsets (groups) resulting from of the decision tree. Table 2 suggests, Most of such groups are small. So we could build efficient classifiers. In practice, we could use the association method (section 3) to make the groups even smaller. For example, when class 49 is present at the upper zone, only the classes 2, 3, 4, 5 and 44 are candidates from Group A.

Let us now present the decision tree model in Fig. 3. Each criterion is given a number and described below.

1. This criterion checks if any lobe exists in the text symbol.
2. This criterion checks the existence of vertical line (VL) in the text symbol. The VL is found by applying morphological operations with linear structuring element having 90% height of the concerned component image.
3. This criterion compares if the lobe width is less than a threshold $T_1 = 7$.
4. Eliminating the upper zone, the text symbol ই results only a lobe. By applying a threshold $T_2 = 0.88$ this criterion separates out ই .

Table 2. The groups obtained from decision tree

Group	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
Class	2-5, 8, 9, 18, 24, 32, 42, 44, 45	20	22	17	15, 39	7, 30	21, 26	37	10, 19	43	6	13, 23, 35, 36, 38, 40	16	27, 31, 41	11	29	12, 14, 25, 34	1	28, 33

5. This criterion finds text symbols (e.g. Group L) with rightmost VL.
6. Some text symbols (e.g. Group N) has a wide lobe comparable to the width of the component. This criterion decides if the lobe width is only a threshold ($T_3 = 1.35$) less than the component width.
7. Using a threshold $T_4 = 18$ indicating component height, the lobe may be vertically at the lower or upper halves. Two separate groups can be formed.
8. We trace the contour of the VL, and check if exactly one other component is connected with the VL. To cope perturbations, such a component must have at least a 20 pixel run, connected with the VL. We further use the term "connectivity" to denote number of components connected with the VL.
9. Certain groups can be formed if left connectivity is one.
10. We check the number of VLs in this criterion. The second VL is, however, smaller than the first. So, we use a shorter (80% of component height) structuring element.
11. Group J is only a VL. So this is sufficient criterion to separate.
12. Only one component (CC) is connected with VL. We trace the boundary contour on which we meet the first pixel of this component. This criterion checks if the position of this pixel is only a threshold ($T_5 = 0.5$) high from the bottom of the VL.
13. Some components has solid lobes (e.g. Group R, S). To separate those components this is the correct criterion to be checked.
14. With $T_6 = 7$ we determine the position of the solid lobe.

5   Results and Discussions

Our dataset consists of 200 images captured by a digital still camera (14.1MP). Along with it we also have 50 scanned document images. First, we present the text extraction results. Consider the "X-Ray" image (Fig. 4(a)) as an example. The headline (Fig. 4(d)) attached components are shown in Fig. 4(e). Comparing this with Fig. 4(a), we notice that all but three text symbols are present. Next, after eliminating boundary attached components we can remove blob at the top of the image (Fig. 4(f)). The ER and counting the number of lobes have no effect on this image. Further, we can filter out the box like non-text, surrounding the text portion, by testing the aspect ratio (Fig. 4(g)) and the arrow like components using ratio r (Fig. 4(h)). Afterwards, Fig. 4(i) gives the results of the identified headline unattached text symbols. It may be noted that the text symbols absent in Fig. 4(e) are now identified successfully. Finally, the symbols after removing the headlines are shown in Fig. 4(j). Note that we have only
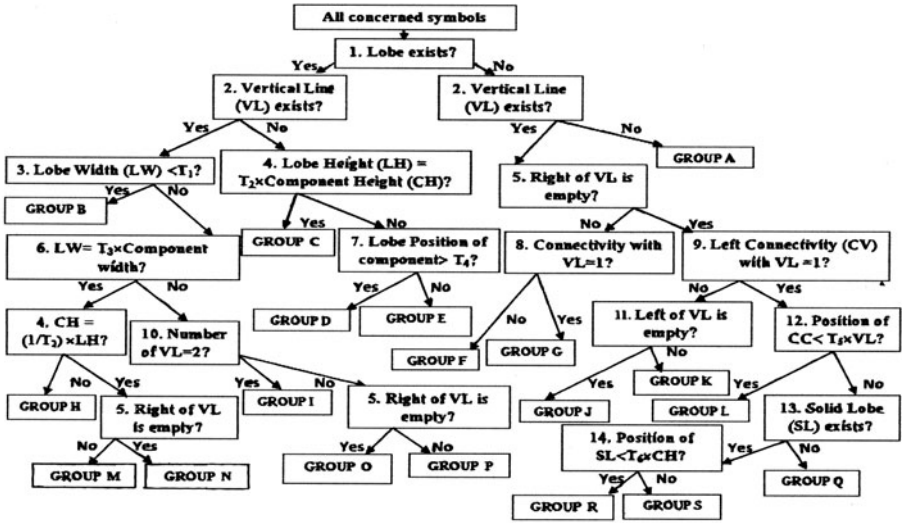
Fig. 3. The decision tree for middle zone symbol recognition
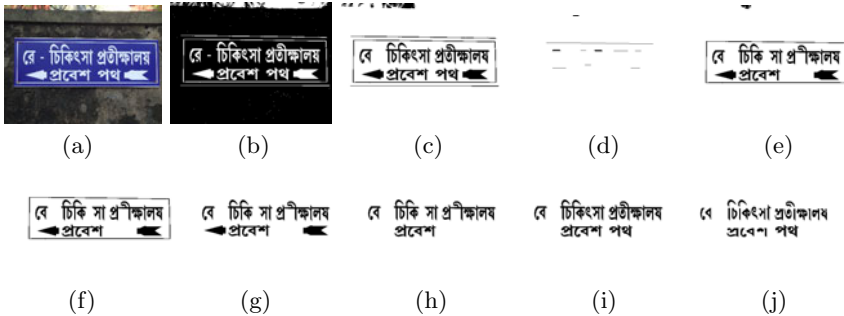
Fig. 4. (a) Input "X-Ray" image, (b) binary image, (c) after removing too small and large components, (d) the headlines and (e) attached components, (f) after removing boundary attached components, (g) after testing aspect ratio, (h) after testing ratio r, (i) headline unattached components, and (j) after removing the headlines
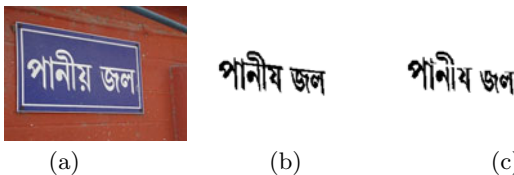
Fig. 5. (a) Input "Water" image, (b) its text portion, and (c) after headline removal

⌢ and ↷ text symbols in the upper zone. Moreover, this particular image has compound characters ঝ and ক্ষ . Such characters are beyond our present study. In the next example, we highlight an important aspect of our dataset. In Fig. 5(a), we show the "water" image. This image has perspective distortion. Though we do not apply any perspective correction procedure, our algorithms extract text symbols successfully (Fig. 5(b)). The headline removal also works satisfactorily as presented in Fig. 5(c). The precision and recall values of the text extraction algorithm based on 250 images, are respectively 70.7% and 73.3%.
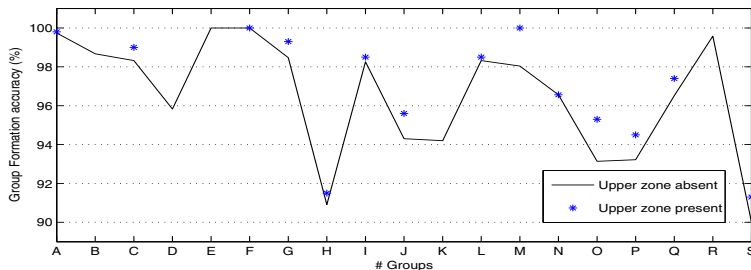
Fig. 6. Group formation accuracy depending presence/absence of upper zone

We now apply the work of [7] to all middle-lower zone text symbols. Besides, the upper zone symbols are recognized separately, using MLP. The lower zone classes 46, 47 and 48 give recognition accuracies 98.56%, 96.47% and 97.33% respectively. On the other hand, for the upper zone classes 49, 50, 51 and 52 the recognition accuracies from MLP are 99.33%, 97.56%, 98.67% and 99.5% respectively. In Fig. 6, we describe how accurately a group is formed depending upon the upper zone is absent/present. Note that the presence of upper zone makes most groups to form more accurately. The groups are produced by the decision tree, so Fig. 6 measures the reliability of the decision tree. Finally, we put the recognition accuracy of all the middle zone symbols in Table 3. This results include the error produced by all previous operations. Note that accuracy is higher if some upper zone symbol is present. The association reduces the size of the groups even to a singleton. Since the MLP in upper zone performs well, the final recognition becomes satisfactory.

6   Summary and Future Scope

This article proposes a recognition scheme of Bangla text symbols embedded in outdoor natural scene images. Our method introduces a decision tree classifier based on structural and topological features. This tree produces small symbol groups, well recognized by MLP. A script based boosting is proposed to improve recognition accuracy. The proposed scheme is very efficient in terms of time and is extendable to printed scanned documents. Future studies may aim at the use of probabilistic scoring at the internal nodes instead of a deterministic decision.

Table 3. Recognition (in %) of the middle zone text symbols

Text class	Some symbol at upper zone	No symbol at upper zone	Text class	Some symbol at upper zone	No symbol at upper zone	Text class	Some symbol at upper zone	No symbol at upper zone	Text class	Some symbol at upper zone	No symbol at upper zone	Text class	Some symbol at upper zone	No symbol at upper zone
1	-	99.6	10	-	91.7	19	98.2	88.5	28	90.2	89.3	37	91.5	90.9
2	99.7	97.5	11	95.3	93.1	20	-	98.7	29	94.5	93.2	38	97.2	96.7
3	99.7	-	12	-	84.5	21	98.7	97.9	30	100	99.3	39	-	99.7
4	93.7	90.1	13	90.7	90.7	22	99.0	98.3	31	88.3	88.3	40	98.2	97.0
5	98.3	-	14	-	86.3	23	97.4	95.7	32	96.4	93.3	41	82.1	82.1
6	-	94.2	15	-	100	24	96.1	92.5	33	90.6	87.5	42	-	99.1
7	-	99.7	16	100	98.0	25	97.0	86.2	34	-	87.3	43	95.6	94.3
8	-	95.7	17	-	95.8	26	98.4	97.3	35	97.4	96.2	44	99.3	98.5
9	-	98.3	18	98.7	92.3	27	87.4	87.4	36	97.4	97.2	45	-	95.4

References

1. Jung, K., Kim, I.K., Kurata, T., Kourogi, M., Han, H.J.: Text scanner with text detection technology on image sequences. In: Proc. of Int. Conf. on Pattern Recognition, vol. 3, pp. 473–476 (2002)
2. Liang, J., Doermann, D., Li, H.: Camera based analysis of text and documents: a survey. Int. Journ. on Doc. Anal. and Recog (IJDAR) 7, 84–104 (2005)
3. Bhattacharya, U., Parui, S.K., Mondal, S.: Devanagari and bangla text extraction from natural scene images. In: Proc. of the Int. Conf. on Document Analysis and Recognition, pp. 171–175 (2009)
4. Pal, U., Chaudhuri, B.B.: Indian script character recognition: A survey. Pattern Recognition 37, 1887–1899 (2004)
5. Roy, A., Parui, S.K., Paul, A., Roy, U.: A color based image segmentation and its application to text segmentation. In: Proc. of Ind. Conf. on Computer Vision, Graphics & Image Processing, pp. 313–319 (2008)
6. Chaudhuri, B.B., Pal, U.: A complete printed bangla ocr system. Pattern Recognition 31, 531–549 (1998)
7. Parui, S.K., Bhattacharya, U., Datta, A., Shaw, B.: A database of handwritten bangla vowel modifiers and a scheme for their detection and recognition. In: Proc. of Workshop on Computer Vision Graphics and Image Processing, pp. 204–209 (2006)
8. Bhowmik, T.K., Ghanty, P., Roy, A., Parui, S.K.: Svm-based hierarchical architectures for handwritten bangla character recognition. Int. Journ. on Doc. Anal. and Recog (IJDAR) 12, 83–96 (2009)

Learning Global and Local Features
for License Plate Detection

Sheng Wang[1], Wenjing Jia[1], Qiang Wu[1], Xiangjian He[1], and Jie Yang[2]

[1] Research Centre for Innovation in IT Services and Applications (iNEXT)
University of Technology, Sydney, Broadway 2007, Australia
[2] Lab of Pattern Analysis and Machine Intelligence
ShangHai Jiao Tong University, Shang Hai 200240, China

Abstract. This paper proposes an intelligent system that is capable of automatically detecting license plates from static images captured by a digital still camera. A supervised learning approach is used to extract features from license plates, and both global feature and local feature are organized into a cascaded structure. In general, our framework can be divided into two stages. The first stage is constructed by extracting global correlation features and a posterior probability can be estimated to quickly determine the degree of resemblance between the evaluated image region and a license plate. The second stage is constructed by further extracting local dense-SIFT (dSIFT) features for AdaBoost supervised learning approach, and the selected dSIFT features will be used to construct a strong classifier. Using dSIFT as a type of highly distinctive local feature, our algorithm gives high detection rate under various complex conditions. The proposed framework is compared with existing works and promising results are obtained.

Keywords: AdaBoost, Intelligent system, License plate.

1 Introduction

License Plate Recognition (LPR) has been widely adopted into numerous applications such as traffic law enforcement, unattended parking, security control and stolen vehicle verification. License Plate Detection (LPD), as the most crucial step in a LPR system, has attracted the attention of many researchers. Factors contributing to the complexity of LPD systems include but are not limited to moving camera (causing degenerated image quality and varied viewing angles), changing illumination, customized license plates (with various sizes, aspect ratios, colors, fonts and formats) and complex background.

LPD methods proposed in recent years can be divided into two categories, methods based on template matching techniques and methods based on machine learning algorithms.

In [1], with a pre-defined threshold value, vertical edge density image is filtered using a license plate pattern match filter. Locations of license plates can then be found from the filter output. In [2], Hue, Saturation, and Intensity (HSI)

color features are used to identify potential license plate regions before geometrical properties of license plates are used to classify those potential license plate regions. In [3], candidate license plate regions are obtained based on clustering Harris corners. These candidate regions are then classified based on statistical features. [1], [2] and [3] can be categoried into template matching based methods.

In [4], Zhang *et al.* proposed a six-layer cascade constructed by thresholding global statistical features and boosting local Haar-like features. In [5], AdaBoost Classifier based on Haar-like features are used in concatenation with Support Vector Machine (SVM) classifier based on Scale-invariant Feature Transform (SIFT) features to increase the performance of the license plate detector. In [6], multiple features based on both saliency features (including character saliency, window saliency and entropy saliency) and non-saliency features (such as character quantity, overlap, aspect ratio and location) are combined with a Bayes Classifier to produce classification results. [4], [5] and [6] are learning based methods.

In this paper, we are going to follow the cascaded framework proposed by [4], where a two-stage cascade framework is construced based on both global feature and local feature. However, differently from [4], we use features that are more salient and our experimental results prove that using these features, better performance can be achieved with a lot fewer features needed. As a result, the time needed to train a classifier that is comparable to (in fact better than) [4] has been greatly reduced, and details will be reported in Section 3.

Compared with the method reported in [4], our contribution can be summarised as follows.

Firstly, compared to [4], the global features that we use are more salient. Not only the global feature descriptors themselves are more discriminative, but also the classifier has been enhanced. With more details, a Bayes Classifier is used to substitute the classifier based on a threshold value and a pairwise histogram distance measurements is used to substitute the distance of the averaged gradient magnitudes.

Secondly, with a higher detection rate and a lower false positve rate, the overall complexity of our approach is much less compared to that of [4]. We use dSIFT as weak features in this paper. The dSIFT features are much stronger than Haar-like features, so the number of dSIFT features needed to perform a good classification is far fewer. The details will be reported in Section 3.

Thirdly, the dSIFT features are extracted for multiple grids of different densities on an image. Thus, they can tolerate the changes of scales of the image. The dense grids can locate fine texture information and the sparse grids can locate coarse texture information. Therefore, our approach does not require scale normalisation during training process.

Fourthly, we use the Soft Cascade approach being reported in [7]. Comparing with the traditional Viola-Jones Cascade being used in [4], the Soft Cascade is advantageous.

The rest of this paper is organized as follows. Section 2 introduces the enhanced global features using a Bayesian Model. Section 2 also introduces the

integration of local features using a supervised learning method. Section 3 provides experimental results, and Section 4 concludes this paper.

2   LPD with Cascaded Global and Local Features

2.1   Overview of the Proposed Framework

Figure 1 gives an overview of the proposed framework. The results from the first stage will be used as input for the second stage, where true positive samples are considered as positive samples and false positive samples are considered as negative samples.
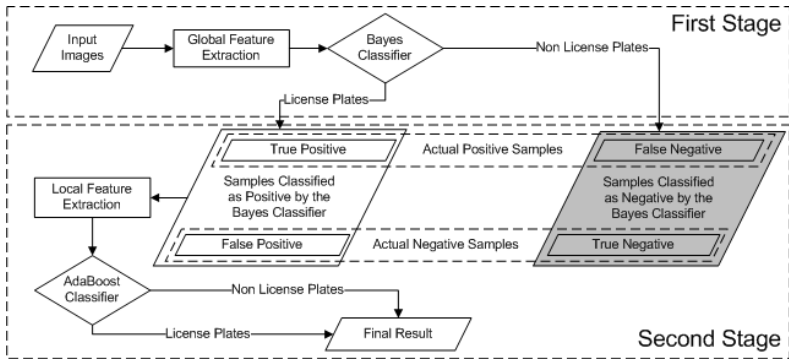
Fig. 1. Overview of the Framework

2.2   Probability Density Estimation of Global Features

As mentioned in Section 1, in this paper, the global features are more salient compared to those being used in [4]. In [4], the average value of gradient magnitude is being used as a primary criteria to measure the resemblance of a detection window with a license plate. In this paper, instead of calculating the average, a histogram will be constructed to better distinguish a license plate from a non-license plate.

Exploring Sub-block Correlations with Histograms. One global feature descriptor as proposed in [4] is represented by

$$D_G = \frac{1}{N} \sum_i \sum_j G(i,j). \tag{1}$$

Another is represented by

$$V_G = \frac{\sum_{i=1}^{n} |g_i - g|}{n \cdot g}. \tag{2}$$

In (1), D_G is in fact the averaged gradient magnitude for the detection window. Also, in [4], the detection window was divided into 12 sub-blocks as illustrated in Figure 2. In (2), g_i is defined as the mean value of the gradient strength at sub-block i, and g is defined as the mean value of the gradient strength of the whole detection window. In fact, g is equal to D_G. n is the total number of sub-blocks.

(a) License Plate   (b) Background

Fig. 2. Dividing the detection window into 12 sub-blocks (Image courtesy of [4])

In this paper, we are going to use a new feature based on correlating the histogram distance features being proposed in [8] as our global feature. There two benefits using histograms, as pointed out by [8]. One is that histograms provide more detailed information, and the other is that with the approach reported in [9], histograms can be computed linearly, which is a precious computational advantage.

Combining the features presented in [4] with those presented in [8], we can construct a new feature based on correlated histogram distances. The new global feature is extracted by first calculating a histogram for each sub-block as illustrated in Figure 2 (with n sub-blocks, we have n histograms), then exploring correlations between histogram distances (for a detection window with n sub-blocks, the number of pairwised histogram distances is C_n^2). In another word, we can construct a $\frac{1}{2} \times n \times (n-1)$-Dimensional global feature vector for each sample. The global feature for a sample with n sub-blocks can be represented by

$$\mathbf{V} = \{D_1, D_2, ..., D_{\frac{1}{2}n(n-1)}\}. \tag{3}$$

Each dimension of the proposed global feature $\mathbf{V}$ in (3) is represented by

$$D = \frac{\sum_{j=1}^{N}(f[j] - g[j])^k}{\sum_{j=1}^{N}(f[j]^k + g[j]^k)}. \tag{4}$$

In (4), N is the total number of bins defined for the histograms. For instance, if we are going to use the pixel intensity values in each sub-block to construct the histogram, for each sub-block, the maximum number of bins of the corresponding histogram is 256. $f[\cdot]$ and $g[\cdot]$ are the histograms calculated from two distinct sub-blocks as illustrated in Figure 2, and $f[j]$ and $g[j]$ corresponding to the $j-$th bins of histograms $f[\cdot]$ and $g[\cdot]$, respectively. Note that here j is corresponding to bin index of the histograms, not the index of the sub-blocks. k is the exponential factor that defines a family of normalization schemes (distances measurements) between any two distinct sub-blocks. In [8], k is equal to two.

Constructing a Bayes Classifier. Based on our observation that those $\frac{1}{2} \times n \times (n-1)$ global features comply with the naive Bayesian assumption, so a Bayes Classifier can be constructed. This classifier will assign a new observation to its most probable class based on its responses of histogram distances. Note that an optimal classifier can be built in a variety of situations where the assumptions are wildly violated [10]. As a typical assumption for continuous data, we assume that those $\frac{1}{2} \times n \times (n-1)$ global features are in Gaussian distribution.

Here, we use Bayes Classifier mainly because compared to the threshold classifier, the Bayes Classifier can give a better classification result based on our experiments. Moreover, the Bayes Classifier is computationally cheap and it gives more flexibility compared to a threshold value based classifier. Details of the experimental results will be reported in Section 3.

Although the Bayes Classifier can give a better classification than that of the threshold classifier in our experiments, it is not powerful enougth to solve the entire problem of identifying the license plates from static images in our case. The classification results of the Bayes Classifier still contain some false positive detection windows. Subsequently, a second stage with local features will be needed.

2.3   Learning Local Features

Dense-SIFT (dSIFT) features were used for gender recognition on face images in [11]. In the paper, the dSIFT features, instead of the SIFT features, were used to address the problems caused by small image sizes, missing texture and poorly illuminated faces. We choose the dSIFT as the local features because, in our case, we have faced the same problems as mentioned in [11].

Compared with the SIFT algorithm, which only extracts scale invariant keypoints from an image, dSIFT algorithm first divides an image into a dense grid, then extracts keypoints from the square patch around each grid point. The number of neighborhood pixels around a grid point is determined by patch size p, which is the length of the square patch. In this way, even for images which are small, with simple texture information, and poorly illuminated, dSIFT can always extract a number of keypoints. As a contrast, SIFT may not be able to extract any keypoints from such images. In dSIFT features, each keypoint corresponds to one grid point. In this paper, we use three different levels of patch sizes. The details will be explained in Section 3.

A sample image of size $M \times N$ can be represented by a dSIFT image of size $M' \times N'$ as shown in (5). Note that for adjacent keypoints having the same patch size, their patches overlap each other by $p \times (p-1)$ pixels, so, $M' = M - p$ and $N' = N - p$.

$$\mathbf{DS} = \{\mathbf{KP}_1, \mathbf{KP}_2, ..., \mathbf{KP}_{M' \times N'}\}. \tag{5}$$

The i-th keypoint $\mathbf{KP}_i$ on the dSIFT image $\mathbf{DS}$ is represented by a 128-D feature vector as:

$$\mathbf{KP}_i = \{d_i^{(1)}, d_i^{(2)}, ..., d_i^{(128)}\}, (1 \leq i \leq M' \times N'). \tag{6}$$

Applying the Linear Discriminant Analysis (LDA) dimensionality reduction, the dimension of the dSIFT feature at each key point is reduced to one [12]. The keypoints on the grid are regarded as weak features, and their corresponding 1-D keypoint descriptor values are used in a simple threshold function for AdaBoost Algorithm to select the most salient weak features.

As we are using three different levels of patch sizes, a pyramid of grids can be built for each training sample. Pyramids of all training samples are fed into an AdaBoost Supervised Learning Algorithm to train an AdaBoost Classifier at the second stage. The AdaBoost Algorithm is used to select the most salient keypoints from pyramids of keypoint grids. Figure 3 gives an illustration.
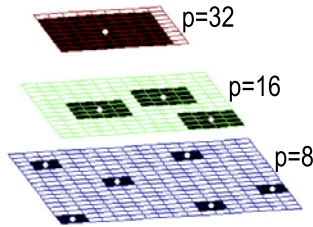
Fig. 3. Selecting most salient keypoints. White dots are the selected keypoints, and black regions are the dSIFT patches for those keypoints.

The local feature being used in this paper is not as weak as the Haar-like feature, so the number of features needed to be selected by the AdaBoost Classifier is considerably fewer, and thus the degree of complexity of the detector can be significantly reduced. Details can be found in Section 3.

3   Experiments

3.1   Improved Global Features

We used 300 positive samples and 300 negative samples to compare the saliency of the new global feature with that of the Density Variance feature. In order to compare the new global feature with the Density Variance feature, we divide the positive samples and negative samples into 12 sub-blocks in the same way as shown in [4] (see Figure 2).

The gradient magnitude histogram of each sub-block is compared with that of every other sub-block. As we have 12 sub-blocks for each sample ($n = 12$), we can construct a feature vector of 66 ($C_{12}^2 = \frac{12 \times 11}{2 \times 1} = 66$) dimensions, so that a Bayes Classifier can be built. Although in theory those 66 features should be independent of each other to construct an optimal classifier, in practice, the overall Bayes Classifier is robust enough to ignore serious deficiencies in its underlying naive probability model [13].

The experimental results listed in Table 1 prove that the Bayes Classifier together with new global feature gives a better classification result than the Density Variance feature with a threshold classifier.

Table 1 gives an overview of the comparison between the new approach with the previous work. As the Density Variance feature is a scalar value, the Bayes Classifier can not be used. In contrast, as the new global feature is a 66 dimensional feature vector, a threshold classifier can not be applied directly. From Table 1, we can see that the new global feature together with the Bayes Classifier can further improve the classification result.

Table 1. Performance Comparison on Global Features

Methods	Min. # of Wrongly Classified Samples
Density Variance + Threshold Classifier	60 out of 600
Correlated Histogram + Bayes Classifier	**47** out of 600

For [4], the Density Variance Feature can be visualized from Figure 4a. In particular, when the threshold value is equal to 0.2236, this classifier gives the optimal performance of 90% accurate classification. The relationship between threshold value and the classification accuracy can be found from Figure 4b.
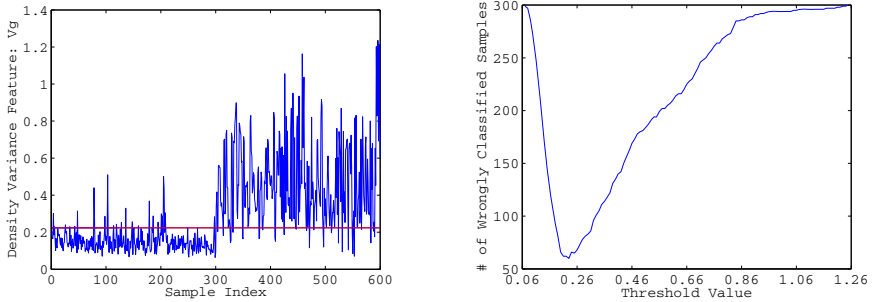
(a) Density Variance Distribution (first 300 are positive samples)

(b) Classification Results vs. Threshold value

Fig. 4. Finding an optimal threshold value

In our case, the optimal value for k defined in (4) is 2.3. Figure 5 gives the detailed information.

One explanation for the high accuracy reported in Table 1 is that the Bayes Classifier can give a correct classification as long as the correct class is more likely than any other classes. Hence, the class probabilities do not have to be estimated very well.
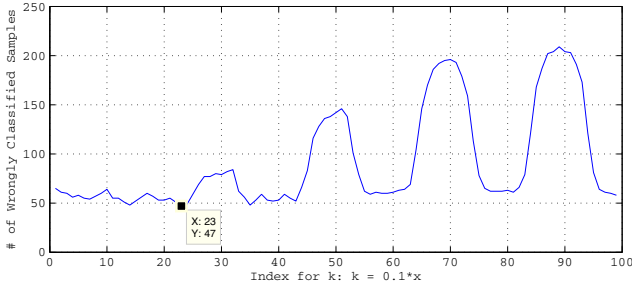
Fig. 5. Classification results according to k

3.2   Stronger Local Features

As mentioned in Section 1 and Section 2.3, the overall complexity of the detector is reduced to a much smaller degree due to the more salient dSIFT feature being used as our local feature. In [4], the total number of weak features needed to construct a strong classifier is approximately 4000, and the number of stages needed for AdaBoost training is at least 10. In addition, for each stage, the nubmer of iterations varies from a few dozens to a few thousands, and in each iteration, one weak feature is needed to be selected from a hundard thousands of Haar-like features. In contrast, in our case, the number of weak features needed, the stages, the number of iterations in each stage and the weak feature pool are all significantly smaller. Details can be found in Table 2.

Table 2. Comparison of Detector Complexity on Local Features

Methods	[4] & [14]	This paper
# of Features Selected	≈ 4000	10
# of Stages Needed	> 10	1
# of Iterations in One Stage	> 3000	10
# of Features Evaluated	> 100000	< 1000

With a much smaller number of salient features selected, the overall complexity of the detector can be greatly reduced. Moreover, with a much smaller number of features being evaluated, the AdaBoost training process can be made much faster.

3.3   Experimental Results

In the experiments, we tested our classifier on the dataset used in [4] and [14]. Of all 460 vehicle images with 474 license plates, 300 images are used for training, the rest 160 images with 169 license plates are used for testing. Some examples of the license plates are shown in Figure 6.

As mentioned in Section 2.3, we use three different levels of dSIFT patch sizes, which are 8, 12 and 16, and the corresponding neighborhood pixels around

Fig. 6. Some examples of the license plates used in our experiments

a keypoint are 64, 144 and 256 respectively. The total number of weak features for AdaBoost algorithm is approximately three times of the sample sizes. The

Table 3. Performance Comparison with Existing Work ⋆

Methods	[4]	[14]	This approach
# of detected license plates	158	163	**162**
# of false positive regions	8	8	**4**

⋆ Dataset: We use the same dataset as used by [4] and [14]. The dataset contains 160 vehicle images with a total of 169 license plates.

experimental results prove that our method can achieve a comparable detection rate and a lower false positive rate compared to previous works.

In [4], the time needed to train a reasonably good detector was approximately 100 hours on a 2.8GHz CPU and OpenCV (Visual Studio C++) environment. In this approach, we can train a comparative detector on a 3.16GHz CPU and Matlab environment within 15 hours [15]. Given that both experiments are conducted on a single thread process, in our case, the computational advantage brought by a faster CPU is limited. Moreover, in terms of compiling and execution speed, Matlab is generally considered as less efficient than C++.

4   Conclusion

In conclusion, we have presented an improved license plate detection system based on the framework given by [4]. Dense-SIFT descriptors are used as Local features and correlated histogram distances are used as global features. The global features together with the Bayes Classifier being constructed can be used to quickly evaluate the detection windows such that only those promising detection windows are subjected to further feature extraction and classification based on dSIFT features and AdaBoost Classifier. Our experiments based on the same dataset as [4] and [14] have demonstrated a comparable detection rate as [14] and a better false positive rate compared to those reported in both [4] and [14].

References

1. Abolghasemi, V., Ahmadyfard, A.R.: A Fast Algorithm for License Plate Detection. In: Qiu, G., Leung, C., Xue, X.-Y., Laurini, R. (eds.) VISUAL 2007. LNCS, vol. 4781, pp. 468–477. Springer, Heidelberg (2007)
2. Deb, K., Jo, K.H.: Hsi Color Based Vehicle License Plate Detection. In: Proceedings of the International Conference on Control, Automation and Systems, Seoul, pp. 687–691 (2008)
3. Luo, J., Yang, S., Guan, R., Niu, H.: A Robust Method for License Plate Detection. In: Proceedings of the Fourth Pacific-Rim Symposium on Image and Video Technology, Singapore, pp. 133–138 (2010)
4. Zhang, H., Jia, W., He, X., Wu, Q.: Learning-based License Plate Detection Using Global and Local Features. In: Proc. ICPR, Hong Kong, pp. 1102–1105 (2006)
5. Ho, W.T., Lim, H.W., Tay, Y.H.: Two-stage License Plate Detection Using Gentle Adaboost and Sift-svm. In: Proceedings of the First Asian Conference on Intelligent Information and Database Systems, Dong Hoi, pp. 109–114 (2009)
6. Lin, K., Tang, H., Huang, T.S.: Robust License Plate Detection Using Image Saliency. In: Proc. ICIP, Hong Kong, pp. 3945–3948 (2010)
7. Bourdev, L., Brandt, J.: Robust Object Detection Via Soft Cascade. In: Proc. CVPR, San Diego, pp. 236–243 (2005)
8. Perrotton, X., Sturzel, M., Roux, M.: Automatic Object Detection on Aerial Images Using Local Descriptors and Image Synthesis. In: Gasteratos, A., Vincze, M., Tsotsos, J.K. (eds.) ICVS 2008. LNCS, vol. 5008, pp. 302–311. Springer, Heidelberg (2008)
9. Porikli, F.: Integral Histogram: A Fast Way to Extract Histograms in Cartesian Spaces. In: Proc. CVPR, San Diego, pp. 829–836 (2005)
10. Lewis, D.: Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398, pp. 4–15. Springer, Heidelberg (1998)
11. Wang, J.G., Li, J., Yau, W.Y., Sung, E.: Boosting Dense SIFT Descriptors and Shape Contexts of Face Images for Gender Recognition. In: Proc. CVPR, San Francisco, pp. 96–102 (2010)
12. Van der Maaten, L.J.P., Hinton, G.E.: Visualizing High-dimensional Data Using T-sne. Journal of Machine Learning Research 9, 2579–2605 (2008)
13. Wikipedia: The free encyclopedia, http://en.wikipedia.org/wiki/Naive_Bayes_classifier
14. Zhang, H., Jia, W., He, X., Wu, Q.: A Fast Algorithm for License Plate Detection in Various Conditions. In: Proc. SMC, Taipei, pp. 2420–2425 (2006)
15. Vedaldi, A., Fulkerson, B.: UCLA Vision Lab, Oxford VGG Lab, http://www.vlfeat.org

Intelligent Video Surveillance System Using Dynamic Saliency Map and Boosted Gaussian Mixture Model

Wono Lee[1], Giyoung Lee[1], Sang-Woo Ban[2], Ilkyun Jung[3], and Minho Lee[1,*]

[1] School of Electronics Engineering, Kyungpook National University,
1370 Sankyuk-Dong, Puk-gu, Taegu 702-701, Korea
[2] Dept. Of Information and Communication Engineering, Dongguk University,
707 Seokjang-Dong, Gyeongju, Gyeongbuk 780-714, Korea
[3] Intelligent Robotics Research Center, Korea Electronics Technology Institute,
401-402, Bucheon Technopark, 193, Yakdae-dong, Wonmi-gu, Bucheon-si, Gyeonggi-do,
420-734, Korea
{wolee,gylee}@ee.knu.ac.kr, swban@dongguk.ac.kr,
mickey3d@keti.re.kr,
mholee@knu.ac.kr

Abstract. In this paper, we propose an intelligent video camera system for traffic surveillance, which can detect moving objects in road, recognize the types of objects, and track their moving trajectories. A dynamic saliency map based object detection model is proposed to robustly detect a moving object against light condition change. A Gaussian mixture model (GMM) integrated with an Adaboosting algorithm is proposed for classifying the detected objects into vehicles, pedestrian and background. The GMM uses C1-like features of HMAX model as input features, which are robust to image translation and scaling. And a local appearance model is also proposed for object tracking. Experimental results plausibly demonstrate the excellence performance of the proposed system.

Keywords: Video surveillance system, Dynamic saliency map, AdaBoost, Gaussian mixture model, Object tracking.

1 Introduction

Due to the increase of crimes and accidents, the needs for video surveillance systems are rapidly increasing. Nowadays, many CCTVs have been installed for monitoring and surveillance. However, most of CCTVs are usually controlled by human operators or fixed at. Hence an automatic intelligent video surveillance system without manual control is necessary to be developed.

In the field of computer vision, many studies for video surveillance have been conducted [1-3]. The Adaboost approach using Haar-like features is one of the state-of-the-art object detection algorithms [4], which is generally implemented using a simple feature extraction method based on many weak classifiers. However this approach is not suitable for detecting various objects and needs large amount training sets. Histogram of oriented gradient (HOG) also shows excellence in detecting

humans [5]. However, the human detection using HOG might be difficult since humans can change their poses, walking and   directions and wearing various clothes. HOG can be simply used to detect humans using a simple classification algorithm such as linear support vector machine (SVM). Instead computational load for this approach is too heavy to implement in a real time embedded system. As well, the traffic surveillance system should be working in real time usually implemented in an embedded system. Thus, a low computation load is considered as an essential property together with high accurate operating performance for the traffic surveillance system.

Therefore, we propose a novel video surveillance system having both a low computation load and a high accuracy. The proposed surveillance system consists of three models such as a moving object detection model, an object classification model and an object tracking model. In order to detect moving objects such as vehicles and pedestrians in a road, a background subtraction method or mixture of Gaussians (MoG) can be used [6] for foreground discrimination from backgrounds. The MoG is one of the most popular algorithms and plausibly copes with multimodal background distributions. In the proposed system, we adopt the dynamic saliency map (dynamic SM) based foreground detection method [7] since the dynamic SM has a robustness against light condition changes occurring in outdoor environments. Moreover, the dynamic SM model successfully pop-outs the moving objects by considering changing dynamics of SM. As the verification step, the system utilizes the global and local features of object images and performs object classification by the boosted Gaussian mixture model (boosted GMM) classifier [8]. This object classification process is considered since the detected foreground may not be the exact object's region and may include background. For the boosted GMM model, C1-like features of HMAX model are used as input features, which are robust to object translation and scaling. The boosted GMM model classifies vehicles, pedestrians and backgrounds.

Moreover, we propose a novel tracking approach based on a local appearance model for successfully tracking of moving objects, which is needed to play a role for locally detecting objects in various situations such as stop and/or occlusion of moving objects. The local appearance model also plays an important role for adaptively tracking objects in case moving objects' shapes and sizes are changed during moving on the road. For satisfying these purposes, the local appearance model is adopted for an object tracking method among many well known tracking algorithms.

This paper is organized as follows; in section 2, we describe the proposed video surveillance system. In section 3, experimental results and discussion will be described. Finally, conclusion and further works are discussed in section 4.

2   Proposed System

The proposed system consists of three main steps; moving object detection, object recognition and object tracking. In the moving object detection part, the system localizes moving object candidate regions by a dynamic saliency map. Then the system identifies whether the regions are interesting objects such as vehicles and pedestrians or not. Finally, object tracking is performed to track moving objects and improve interesting object detection rate.

2.1   Moving Object Detection

The proposed moving object detection model is based on the dynamic SM based detection model [7]. The dynamic SM is based on successive static SMs obtained from a time sequence of input scenes. The entropy through time is used to analyze the dynamic characteristics of successive static SMs. With respect to the first frame at time, τ, the most appropriate scale X_s for each area centered at location X is obtained which considers spatial dynamics at each location. The probability mass function of a dynamic scene $PH_{D(X_s,X,\tau+n\Delta t)}$ is obtained from the histogram of the entropy values for a sequence of static SMs in $(n+1)$ frames from τ to $\tau + n\Delta t$ where n is the number of continuous frame and Δt denotes the sampling time. The entropy value at location X is calculated from the histogram of pixel values of the local area centered at X with size Xs in a static SM. By using the probability mass function for a dynamic scene, the time varying entropy is calculated using Eq. (1).

$$T_D(X_s,X,\tau+n\Delta t) \equiv -\sum_{d\in D} PH_{D(X_s,X,\tau+n\Delta t)} \log_2 PH_{D(X_s,X,\tau+n\Delta t)} \tag{1}$$

The entropy value T_D at each pixel X represents a fluctuation of visual information with time, from which this model generates a dynamic SM. And a dynamic SM is normalized between 0 and 1 for candidate localizing process. To localize a moving object candidate area, the dynamic SM is binarized with a threshold. After isolating blobs in the binarized dynamic SM using morphological closing, moving object candidates are localized using a labeling method.

2.2   Object Recognition

The object candidate regions detected by the moving object detection model may be translated or not exactly fitted to the real object regions. Therefore an object recognition module is implemented using a GMM, in which the C1 features of the Hierarchical MAX (HMAX) model [9] are used as the input features. The C1 features have scale and translation invariant property. Each GMM is established for each object class and the greedy learning algorithm and partial EM are considered to train the GMM [10]. Moreover, the Adaboost algorithm is applied to the GMMs to achieve a higher accuracy of classification [11].

In the proposed model, we use C1-like features based on the C1 features. The main difference is that we use edge orientation information based on the Sobel operator instead of using Garbor filters, which can reduce the computation time. The edge orientation features are relevant to the S1 features of the HMAX model.

As a learning algorithm of a GMM, the EM algorithm is generally used [12]. The EM algorithm is known to converge to a locally optimal solution. However, it does not guarantee a globally optimal solution. Therefore we use the greedy learning algorithm with the partial EM searches for efficient learning. After training a GMM for an object, the GMM can estimate the probabilities for the inputs. A higher probability means that the input is likely to belong to the learned class. Since the learning of each GMM is performed for each different class, the GMM by the specific class learning may have poor classification accuracy when the objects in different

classes are very similar to each other. To solve this problem, we applied the Adaboost algorithm to efficiently collect each GMM for constructing a strong classifier. As a weak learning algorithm for the Adaboost algorithm, we built a simple classifier using a component of GMMs with a threshold which is determined experimentally.

Thus, we construct the clusters of boosted GMM classifiers for 2 object classes such as vehicles and pedestrians. Each cluster has 4 boosted GMM classifiers of 3 global feature bands and 1 local feature. Outputs of boosted GMM classifiers for each object class are summed and the classification result is obtained by the maximum value between clusters. If outputs of boosted GMM classifiers are lower than a threshold, the classification result is considered as backgrounds. Accordingly, the proposed boosted GMM module classifies three object classes such as vehicles, pedestrians, and backgrounds.

2.3   Object Tracking

The dynamic SM based detection model is designed to detect moving objects. It cannot detect still objects because still objects do not have dynamic characteristics. Thus we solve this problem with the object tracking algorithm with the local appearance model. Objects cannot be created in the middle of the road but move in from other places. Hence the proposed system tracks objects entering to and leaving the monitored scene.

2.3.1   Local Appearance Model

To track objects the proposed system utilizes the local appearance feature from an object image. As features, we use a color histogram to represent the color appearance and the cell matching score to represent the shape of the object. The color histogram has 24 bins (8 for each RGB channel) and to avoid the aliasing, adjacent bins are increased by 0.5 when the center bin is increased by 1. The color histogram is normalized by Eq. (2).

$$hist'(i) = \frac{hist(i)}{\sum_i (hist(i))^2} \qquad (2)$$

The correlation of two histograms is calculated by the Pearson product moment correlation coefficient equation shown in Eq. (3).

$$\gamma_{xy} = \frac{n \sum hist'_x(i)hist'_y(i) - \sum hist'_x(i) \sum hist'_y(i)}{\sqrt{n \sum hist'^2_x(i) - (\sum hist'_x(i))^2} \sqrt{n \sum hist'^2_y(i) - (\sum hist'_y(i))^2}} \qquad (3)$$

where n is the dimension of vectors (n is 24). Also x means an object region in a previous frame and y means an object region in a current frame.

To compute the cell matching score, we convert the object image to gray scale and divide into several sub regions, in our implementation, the number of sub-regions is 6×8. The procedure is described in Fig. 1. In each sub region, mean and variance of pixels are obtained. Comparing mean and variance values with the other object image, we can set matching cells by Eq. (4).

$$m\_cell_{xy}(i, j) = \begin{cases} 1, & \text{if } |mean_x(i, j) - mean_y(i, j)| < \theta_m \\ & \text{and } |var_x(i, j) - var_y(i, j)| < \theta_v \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

The cell matching score, m_score is computed by Eq. (5) and the similarity S between two object regions is computed by Eq. (6). The proposed system creates the similarity matrix between objects in previous frame and current frame and tracks objects by finding maximum S and linking.

$$m\_score_{xy} = \frac{\sum\limits_{i,j} m\_cell_{xy}(i, j)}{Total\_number\_of\_cells} \tag{5}$$

$$S_{xy} = (m\_score_{xy} + \gamma_{xy})/2 \tag{6}$$

The local appearance model is also utilized for detecting objects. When an object is moving it can be detected but when it stops moving temporally it disappears. Therefore the proposed system scans to detect unlinked objects near their locations and can find stopped objects or temporally occluded objects. The scanning method is a sliding window around the object's location and the detected position of the object is the position of maximum S and the size is the same as the previous size.
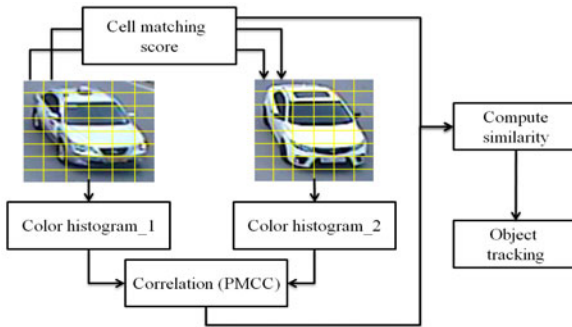
Fig. 1. The procedure of computing the similarity between two objects

3 Experimental Results

To evaluate the proposed system, we ran the system on our test bed. In the test bed, there are three IP cameras. Each camera captures different points of view but they aim at the same intersection. We also use our video datasets recorded under multiple circumstances. The datasets include three different weather videos and successive 800 frames of each dataset are used to evaluate the detection performance. Fig. 2 shows examples from our dataset and table 1 shows the experimental result of the dynamic SM based object detection model in different datasets.

Fig. 2. Examples of the datasets for evaluating the detection model

Table 1. The experimental results of the proposed moving object detection model

	# of total objects	Detection accuracy	False alarm rate
Video dataset 1	1149	93.04%	22%
Video dataset 2	1681	90.12%	47%
Video dataset 3	1092	94.96%	21%

Dataset 2 is the worst in the whole dataset, because it included many complex scenes with strong shadow effects. Furthermore, dataset 1 was recorded in rainy conditions. However, the moving object detection model shows a high accuracy of more than 90% in all the datasets. Although those results show a few false alarms of more than 20%, the proposed system can discard the unexpected candidates by a posterior process using the object identification model.

To test the object identification performance, we use official object database including CBCL vehicles, CBCL pedestrians, Caltech vehicles, Daimler pedestrians and some background images collected from Caltech database [13-14]. We also tested for object images from the ABR dataset which was recorded at our institution. In each dataset, 300 object images per class are used to train the recognition model (100 for creating a codebook, 200 for training a classifier) and 200 images per class are used to measure classification accuracy of the proposed model. Table 2 shows the comparison of experimental results of the proposed model and our previous model for the datasets [15]. The proposed model has similar accuracy for vehicles and pedestrians, but higher performance for background images on average. Moreover, computational speed is much faster than the previous model, in which the proposed model takes 64msec on average, while the previous model takes 137 msec for the 78x78 size object images (Intel Core2Quad 2.4GHz).

We use the CLEAR MOT metrics for evaluating object tracking performance of the proposed model [16]. Videos from the three IP cameras are used and each video has about 1800 successive frames. Although the view-points and moving directions of objects are various as shown in Fig. 3, the proposed tracking algorithm works well and the precision score (intersection of overlapped bounding boxes) and accuracy score (false-negative, false-positive and number of ID switches) are shown in table 3.

In testing experiments, up to four objects simultaneously appear in an input scene and the proposed surveillance system operates in approximately 10 frames per second (Intel Core2Quad 2.4GHz).

Table 2. The experimental results of the proposed object identification model

		Vehicles	Pedestrians	Background
Object dataset 1	Proposed Model	98.5%	99.5%	88%
	Previous Model	98.5%	99%	75.5%
Object dataset 2	Proposed Model	97.5%	99%	74.5%
	Previous Model	100%	94.5	79.5%
Object dataset 3	Proposed Model	96%	98.5%	72%
	Previous Model	99%	97%	62.5%

Table 3. The experimental results of the proposed object tracking model

	Prec.	Accur.	F. Neg.	F. Pos.	ID Sw.
IP camera 1	81.8%	75.7%	22.6%	1.7%	3
IP camera 2	78%	70%	22.9%	7.1%	1
IP camera 3	76.7%	76.2%	17.2%	6.6%	13

Fig. 3. The examples of multi object tracking

4   Conclusion

We proposed a series of computer vision algorithms for a video surveillance system. Since the system was developed for implementation in an embedded system, we consider the trade-off between accuracy and computational load. The dynamic SM based moving object detection model is robust to change of environment and noise. And the object identification model using the global/local feature and the boosted GMM shows high performance on multi-class object recognition problem. The proposed object tracking mechanism makes the system complete, compensating for the limitation of the detection algorithm. And the tracking algorithm enables the function of the system to be extended, for example, to accident detection.

In future work, we plan to improve the tracking performance and develop an incremental learner for multi-class object recognition model.

References

1. Dedeoglu, Y.: Moving Object Detection, Tracking and Classification for Smart Video Surveillance, Ph.D. Thesis, Department of Computer Engineering, The Institute of Engineering and Science of Bikent University (2004)
2. Foresti, G.L.: Object recognition and tracking for remote video surveillance. IEEE Trans. on Circuits and Systems for Video Technology 9(7), 1045–1062 (1999)
3. Wang, J., Bebis, G., Miller, R.: Robust Video-Based Surveillance by Integrating Target Detection with Tracking. In: IEEE Workshop on Object Tracking and Classification Beyond the Visible Spectrum in conjunction with CVPR, pp. 137–142 (2006)
4. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proc. of IEEE Computer Society Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 511–518 (2001)
5. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, vol. 1, pp. 886–893 (2005)
6. Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. In: Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, vol. 2, pp. 246–252 (1999)
7. Woo, J.-W., Lee, W., Lee, M.: A Traffic Surveillance System Using Dynamic Saliency Map and SVM Boosting. Int. J. of Control, Automation, and Systems 8(5), 948–956 (2010)
8. Lee, W., Lee, M.: A Multi-class Object Classifier Using Boosted Gaussian Mixture Model. In: Wong, K.W., Mendis, B.S.U., Bouzerdoum, A. (eds.) ICONIP 2010, Part I. LNCS, vol. 6443, pp. 430–437. Springer, Heidelberg (2010)
9. Riesenhuber, M., Poggio, T.: Hierarchical Models of Object Recognition in Cortex. J. Neurosci. 2, 1019–1025 (1999)
10. Verbeek, J., Vlassis, N., Krose, B.: Efficient Greedy Learning of Gaussian Mixture Models. Neural Computation 15, 469–485 (2003)
11. Freund, Y., Schapire, R.E.: A Short Introduction to Boosting. The Japanese Society for Artificial Intelligence 14(5), 771–780 (1999)
12. Blimes, J. A.: A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. Technical Report ICSI-TR-97-021 (1997)
13. Center for Biological & Computational Learning. MIT, http://cbcl.mit.edu
14. Computational Vision Lab., Caltech, http://www.vision.caltech.edu
15. Woo, J.-W., Lim, Y.-C., Lee, M.: Obstacle Categorization Based on Hybridizing Global and Local Features. In: Leung, C.S., Lee, M., Chan, J.H. (eds.) ICONIP 2009. LNCS, vol. 5864, pp. 1–10. Springer, Heidelberg (2009)
16. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: the CLEAR MOT metrics. Journal on Image and Video Processing 2008, 1–10 (2008)

Contour-Based Large Scale Image Retrieval

Rong Zhou and Liqing Zhang

MOE-Microsoft Key Laboratory for Intelligent Computing and Intelligent Systems,
Department of Computer Science and Engineering,
Shanghai Jiao Tong University, Shanghai, 200240, China
rongzhou@sjtu.edu.cn, zhang-lq@cs.sjtu.edu.cn

Abstract. The paper presents a contour-based method for large scale image retrieval. With the contour saliency map of the object, it could address the shift-invariance problem, and with hierarchical and multi-scale feature extraction, it is able to deal with the scale-invariance problem to a certain extent. Different from existing algorithms, the features used in the retrieval system contain not only local information, but also global information of the object. By taking advantage of this characteristic, we could build a hierarchical index structure which helps to fast retrieval of the large scale database. Furthermore, our method allows two kinds of query image: a hand-drawn sketch or a natural image. Thus it is possible to refine the search results by choosing one image from the list of previous sketch retrieval results as the new query. It brings the better interactive user experiment and the convenience for those who aren't good at drawing. The experiment results verify the performance of our method on a database of four million images.

Keywords: shift-invariance, contour saliency map, hierarchical structure, global-to-local feature, orientation information.

1 Introduction

Contour is a very important channel for human being to recognize or distinguish the objects from an image or a scene. Image retrieval based on contour has been attracted great attention in the data mining society [1], but most of works mainly dealt with image retrieval in small database [2][3][4]. And in large scale database, Eitz [5] presented a method that divides an image into a fixed number of cells, and each cell corresponds to a structure tensor descriptor which stores the main direction of the gradients of the cell. Different from Eitz's method which hasn't index structure and must scan the whole database for each query, Cao [6] presented an index-able oriented chamfer matching method. But both their works have the same limitation that the shift-invariance problem still exists in their retrieval system. The objects in the query image and in the retrieved image must have the same position, this property will reduce dramatically the recall rate in image retrieval. And in most of situation, the users usually only

mind whether they could search the object they wish, and don't care where it is in the image.

To address this issue, we propose a shift-invariance method for large scale image retrieval. It comes from the fact that when human beings see an image, they usually look through the whole image for a short while and then focus their eyes on the salient place of the image, as shown in Fig. 1(a). That means in most of time, people only pay attention to a local part of an image instead of the whole image. So different from existing algorithms, we don't extract features on the whole image, instead, we first find the saliency map of the object which is usually a local part of the image, and then extract features on the part.

(a) Image processing of human visual system.

(b) Framework of feature extraction.
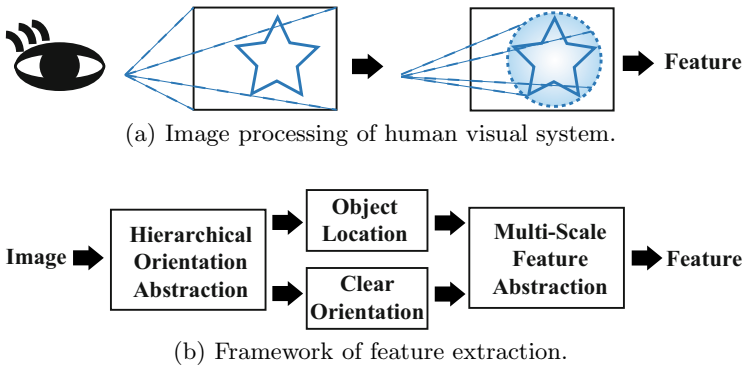
Fig. 1. Framework of feature extraction by simulating human visual system

Another contribution is a contour-based image retrieval prototype system for the database including more than four million images. With hierarchical and multi-scale feature extraction, we could easily obtain not only the position of the object but also the global-to-local orientation features, which brings two advantages: shift-invariance and scale-invariance to a certain extent. These cannot be achieved by most existing retrieval systems. Moreover, the system provides users two query methods: a hand-drawn sketch or a natural image, as shown in Fig. 2. If you are a good painter, you could draw a sketch whatever you imagine, but if the sketch doesn't like what you imagine very much, you can select a natural image which is most similar to what you wish from the list of retrieval results and then make the second retrieval to achieve satisfactory images.

2   Feature Extraction

Fig. 1(b) demonstrates the basic framework of feature extraction of our method. By simulating hierarchical information processing of human visual system, it could obtain a contour saliency map of the object in an image, and at the same time, it could still extract clear orientation information of an object. With the
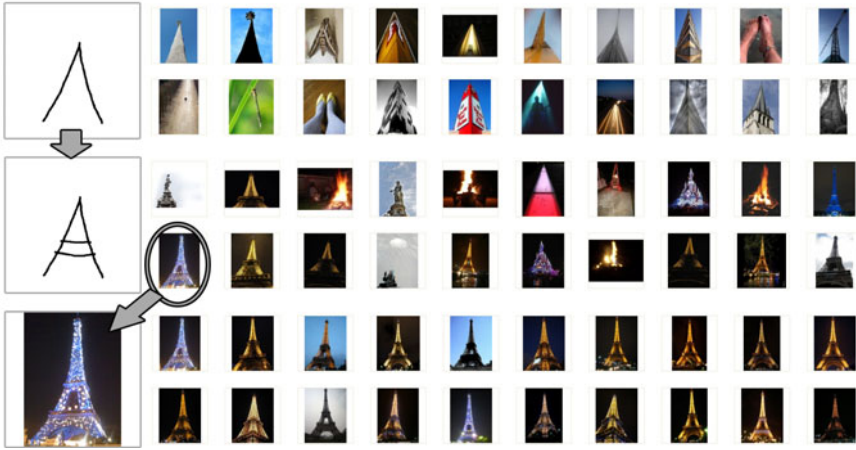
Fig. 2. Illustration of interactive retrieval process. After querying with a hand-drawn sketch, the users could choose one result image as a query and make the next retrieval.
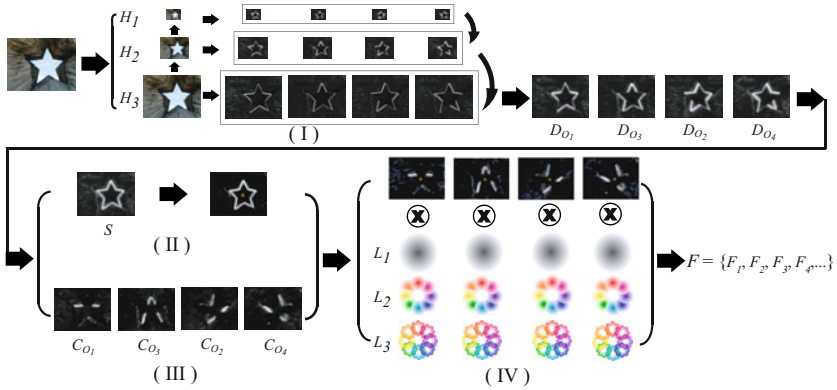
Fig. 3. Details of feature extraction. (I) Hierarchical orientation extraction. (II) Object location. (III) Clear orientation. (IV) Multi-scale feature extraction.

two above mentioned, we can easily know the object's position and contour orientation information. And then, by multi-scale feature extraction, we can obtain the global-to-local feature of the object.

It is well known that human visual system processes the image with hierarchical structure. According to this, hierarchical difference image D_{O_j} and the contour saliency map S are computed from an image as:

$$S = \sum_{j=1}^{N} D_{O_j} = \sum_{j=1}^{N} \left(\sum_{i=1}^{M} [\max_{k} \{D_{H_i O_j}\}]_{m \times n} \right) \tag{1}$$

where $D_{H_iO_j}$ is the difference image of the ith level and the jth orientation, as shown in Fig. 3. And the size of $D_{H_{i-1}O_j}$ is lower than the size of $D_{H_iO_j}$ by two times. k is the red, green, blue color channel, and $\max_k\{\cdot\}$ is the maximum of difference image among the three channels. $[\cdot]_{m \times n}$ means scaling the difference image proportionably to the maximum size m×n. S is the contour saliency map of an image and is normalized to between 0 and 1. In Fig.3, $M = 3$, and $N = 4$, O_j denotes 0, $\pi/4$, $\pi/2$, and $3\pi/4$ orientation respectively. Because the minimum resolution of images for human beings is 32×32 [7], we set the maximum size of $D_{H_1O_j}$ is 32×32, and then m×n is 128×128.

$$S_x = \arg\max_x\{sum(\lfloor S \rfloor_{T_s})_x \star g_x\}, \quad S_y = \arg\max_y\{sum(\lfloor S \rfloor_{T_s})_y \star g_y\} \quad (2)$$

where $\lfloor \cdot \rfloor_{T_s}$ denotes the value greater than T_s, in our experiment, $T_s = 0.25$. $sum(\cdot)_x$ and $sum(\cdot)_y$ are the sum along the axis x and the axis y respectively, g is the Gaussian kernel, and $\star$ denotes convolution. (S_x, S_y) are coordinates of the maximum convolution value in the saliency map, and they denote the centroid of the object in the image.

From Fig. 3 we can see, D_{O_j} cannot represent contour orientation information of the object clearly. Considering 0 and $\pi/2$, $\pi/4$ and $3\pi/4$ are orthogonal respectively, we make the following operation:

$$C_{O_1} = \lfloor D_{O_1} - D_{O_3} \rfloor_0, C_{O_3} = \lfloor D_{O_3} - D_{O_1} \rfloor_0$$
$$C_{O_2} = \lfloor D_{O_2} - D_{O_4} \rfloor_0, C_{O_4} = \lfloor D_{O_4} - D_{O_2} \rfloor_0 \quad (3)$$

where C_{O_j} denotes clear orientation map.

The final feature of an image is:

$$F_{L_pO_jt} = \sum C_{O_j}(x_{L_pt}, y_{L_pt}, r_{L_p}) \cdot G(r_{L_p}) \quad (4)$$

where $F_{L_pO_jt}$ denotes the tth feature of the L_pth level and the O_jth orientation, and $G(r_{L_p})$ is the Gaussian kernel which radius is r_{L_p}, and $C_{O_j}(x, y, r)$ is the region of the clear orientation map which centroid is (x, y) and the radius is r, and $r_{L_p} = 2r_{L_{p+1}}$, $r_{L_3} = 32$. When $p = 1$, $t \in \{1\}$, and when $p = 2$, $t \in \{1, 2, ..., 8\}$, and when $p = 3$, $t \in \{1, 2, ..., 64\}$. So the feature $F = \{F_{L_pO_jt}\}$ has $1 \times 4 + 8 \times 4 + 8 \times 8 \times 4 = 4 + 32 + 256 = 292$ dimensions. And finally, values of the feature are normalized to between 0 and 1.

So the similarity measure of two images is given by:

$$Dist(F, F^{'}) = sim\left(\{F_{L_pO_jt}\}, \{F^{'}_{L_pO_jt}\}\right) \quad (5)$$

where $sim(\cdot)$ could be any similarity measure, for example, Euclidean distance or cosine similarity.

Fig. 4 is the histogram of average value of feature F from 100 thousand images. Region L_1 denotes the first 4 values of F, and region L_2 denotes the following
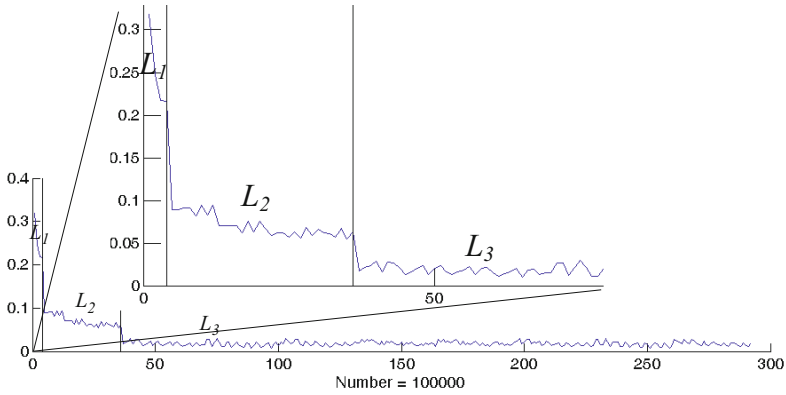
Fig. 4. Histogram of average feature values from 100,000 images. The first 4 values of F belong to region L_1, the first 5 to 36 values of F belong to region L_2, and the rest 256 values of F belong to region L_3.

32 values of F, and region L_3 denotes the rest 256 values of F. L_1, L_2, L_3 are corresponding to L_1, L_2, L_3 in Fig. 3. From Fig. 4, we can see the step-by-step descending trend of F. That is why we call F the global-to-local feature. Values from L_1 to L_3 denote the information which is from global to local respectively, so values in L_1 will occupy a large proportion in distance computing of equation(5). If objects in two images are very different in contour, the difference of values in L_1, L_2, L_3 must be all large, and as a result the similarity score in equation(5) is very low. But if two objects are only a little different, in other word, they should have almost the same global information and are just different in local parts, then only some values' difference in L_3 (maybe still in L_2) is large, but in L_1 must be small, and finally the similarity score in equation(5) is high.

3   Index Structure

Our feature contains an object's global-to-local information, so we select only the first 36 values of F which belong to region $L1$ and $L2$ as shown in Fig. 4 and include most of important information of the object. And for each value, we separate it into some parts, and for each part, there is a corresponding inverted list of images, as shown in Fig. 5. With the index structure, we could select top N1 ($\leqslant T$) candidate results from the database quickly, and then, we select top N2 results from N1 candidate results with similarity measure of first 36 values of F. Finally, we rank the N2 results with similarity measure of all values of F and take them as the finally retrieval results. Thus we could build a hierarchical top-down retrieval structure. In our experiment, $T = 50000$, $N2 = 2000$.
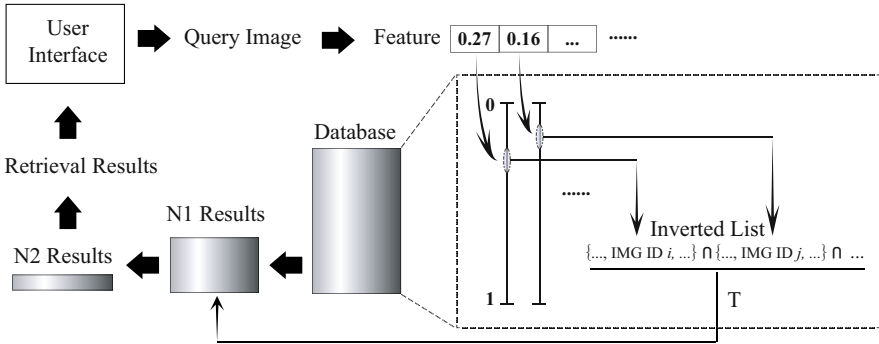
Fig. 5. Index process of database. For each query, N1 results are first selected from 4 million images by inverted files, and then top N2 results are selected from N1 results with similarity measure of first 36 values of F, and final results are from N2 results with similarity measure of all 292 values of F.

4   Experiment

To evaluate our retrieval method, we built a prototype system which database has more than 4 million Flickr images and run it on the server with 2 Intel Xeon 2.4GHz Quad Core processors and 8GB memory. Because the feature of an image has only 292 dimensions, and it takes less than 2KB memory per image, and the memory cost of our system including features of the database and the inverted file is not more than 7GB in total. So a normal server is powerful enough for our system. The average retrieval time is about between 2 and 3 seconds.
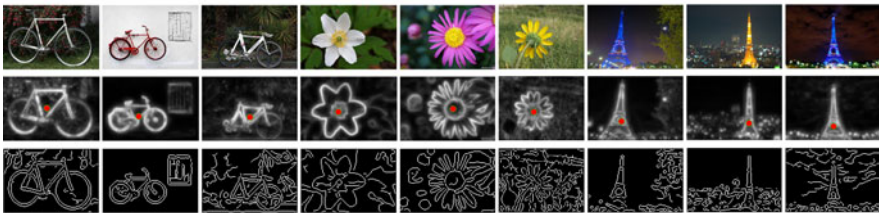
Fig. 6. Some examples of shift-invariance. Top row: the original image. Middle row: corresponding contour saliency map S and the centroid of the object (S_x, S_y) (red point). Bottom row: Canny edge detection.

To better explain why our method could deal with the shift-invariance problem, we display some examples and the corresponding contour saliency maps in Fig. 6. Our method extracts contour information of the object clearly, and further, obtains the centroid of the object. It is hard to be achieved by existing edge detection methods, e.g. Canny edge detector [8], as shown in the bottom
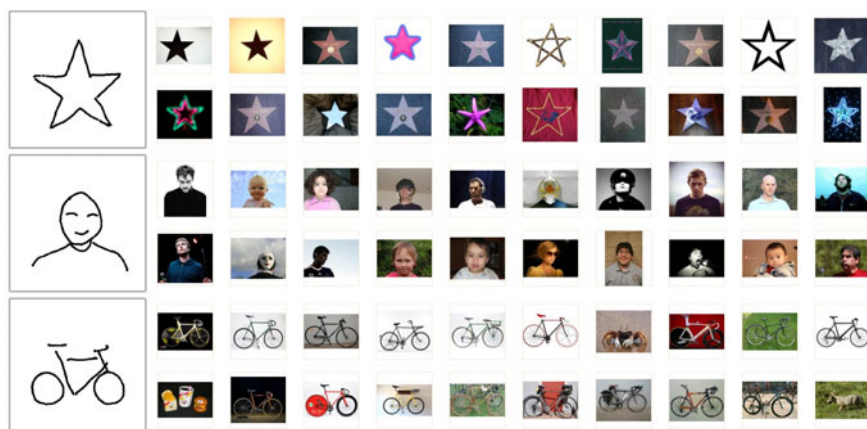
(a) A hand-drawn sketch as a query image
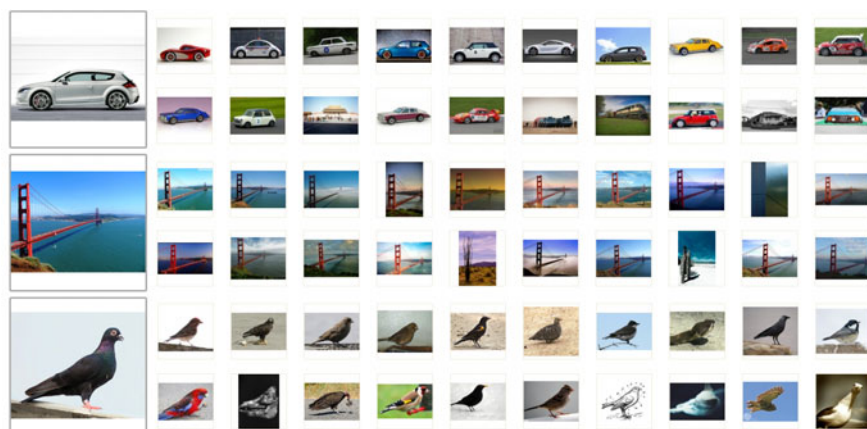
(b) A natural image as a query image

Fig. 7. Some example queries and their top 20 retrieval results from the database of 4 million images

row of Fig. 6. And for most of saliency detection methods [9][10], it is still hard to be achieved. Because existing methods are almost based on information maximization [11], and if there are lots of contours having the same orientation in the image, these contours would be not salient in the saliency map.

Because no existing algorithm or image database is available for us to compare the performance, we just display the retrieval results from hand-drawn sketch and natural image as the query respectively, as shown in Fig. 7. From the results, we can see our method is shift-invariance. And for similar objects with different scales, their proportion of global feature at four orientations would be almost same, thus their similarity score in equation(5) will be high. So our method deals with the scale-invariance problem to a certain extent.

5   Conclusion

We propose a simple and efficient contour-based method for large scale image retrieval. With hierarchical top-down index structure, our method can search the results from 4 million images quickly. Furthermore, it can use not only a hand-drawn sketch but also a natural image as the query image, which brings better interactive query method and the convenience for the users who don't do well in drawing. And our retrieval method is shift-invariance and scale-invariance to a certain extent, which could not be performed by any existing system having been published.

References

1. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image Retrieval: Ideas, Influences, and Trends of the New Age. ACM Computing Surveys 40(2), 1–60 (2008)
2. Belongie, S., Malik, J., Puzicha, J.: Shape Matching and Object Recognition Using Shape Context. IEEE Trans. PAMI 24(4), 509–522 (2002)
3. Tieu, K., Viola, P.: Boosting Image Retrieval. IJCV 56(1/2), 17–36 (2004)
4. Shechtman, E., Irani, M.: Matching Local Self-Similarities across Images and Videos. In: CVPR, pp. 1–8 (2007)
5. Eitz, M., Hildebrand, K., Boubekeur, T., Alexa, M.: An Evaluation of Descriptors for Large-Scale Image Retrieval from Sketched Feature Lines. Computers & Graphics 34, 482–498 (2010)
6. Cao, Y., Wang, C.H., Zhang, L.Q., Zhang, L.: Edgel Index for Large-Scale Sketch-based Image Search. In: CVPR (accepted, 2011)
7. Torralba, A., Fergus, R., Freeman, W.T.: 80 Million Tiny Images: A Large Dataset for Non-Parametric Object and Scene Recognition. IEEE Trans. PAMI 30(11), 1958–1970 (2008)
8. Canny, J.: A computational Approach to Edge Detection. IEEE Trans. PAMI 8(6), 679–698 (1986)
9. Gao, D., Mahadevan, V., Vasconcelos, N.: On the Plausibility of the Discriminant Center-Surround Hypothesis for Visual Saliency. Journal of Vision 8(7), 13, 1–18 (2008)
10. Seo, H.J., Milanfar, P.: Static and Space-Time Visual Saliency Detection by Self-Resemblance. Journal of Vision 9(12), 15, 1–27 (2009)
11. Bruce, N.D.B., Tsotsos, J.K.: Saliency Based on Information Maximization. In: NIPS, vol. 18, pp. 155–162 (2006)

Three Dimensional Surface Temperature Measurement System

Tao Li, Kikuhito Kawasue, and Satoshi Nagatomo

University of Miyazaki, 1-1 Gakuen Kibanadai-nishi,
Miyazaki, 889-2192 Japan
`kawasue@cc.miyazaki-u.ac.jp`

Abstract. In this paper, the three-dimensional surface shape measurement system with the temperature information is introduced. The measurement is established using a three-dimensional surface measurement system and a thermography. The measurement system is composed of CCD camera, a laser and thermography. The laser is projected to the object and the laser streak image appeared on the surface of the object is observed by a CCD camera and a thermography. The streak image recorded by the CCD camera is used to reconstruct the object shape on a computer, and the corresponding temperature data obtained by a thermography is allocated to the reconstructed surfaces of the object on a computer. The obtained data can be used for a quantitative analysis of a heat radiation considering the area and the roughness of the heat source object. Experimental result shows the feasibility of our system.

Keywords: Image processing, Calibration, three-dimensional measurement, Temperature, CCD, Computer vision, Matrices, Thermal-image, Thermography.

1 Introduction

This paper proposes the system that measures the three dimensional shape with its temperature distribution of an object using a CCD and a thermography. A thermography is utilized in various fields such as medical and engineering fields, since it measures the surface temperature of the object instantaneously without contacts to the object. However, the recorded image by the thermography is generally in two dimensional and the quantitative information such as area size and roughness of the heat source object cannot be obtained. As the size of the object appeared in the thermal image is varied depend on the distance of the object from the thermography, qualitative analysis between the temperature and the position cannot be carried out.

In this paper, the three-dimensional thermal-sensing system is introduced. Laser slit ray is projected on the surface of the object. The CCD camera records the streak of the laser appeared on the surface of the object and the thermography records the thermal-image of the object simultaneously. The points on the laser streak are triangulated and the three dimensional position of these points in a world coordinates are calculated [1]-[5]. The arrangement between the CCD camera and thermography are calibrated in advance and the relation between the CCD camera coordinate system and the thermography coordinate system is formulated in the conversion matrices [6]-[8]. Once

the three-dimensional position is measured, the corresponding temperature is allocated from the thermal-image using the conversion matrices. Since the quantitative data of the position with the temperature is measured using our system, the obtained data can be used for an analysis of a heat radiation considering the area and the roughness of the heat source object.

Calibration is an important task in three-dimensional measurement since it influences the measurement accuracy. Generally, calibration process is complicated and is not unified in three-dimensional measurement system. In the proposed system, the relation between CCD coordinates and thermography coordinates have to be determined precisely on the calibration process since it influences the allocation accuracy of the temperature to the surface of the object. The suitable calibration method for a CCD and a thermography is proposed in this paper.
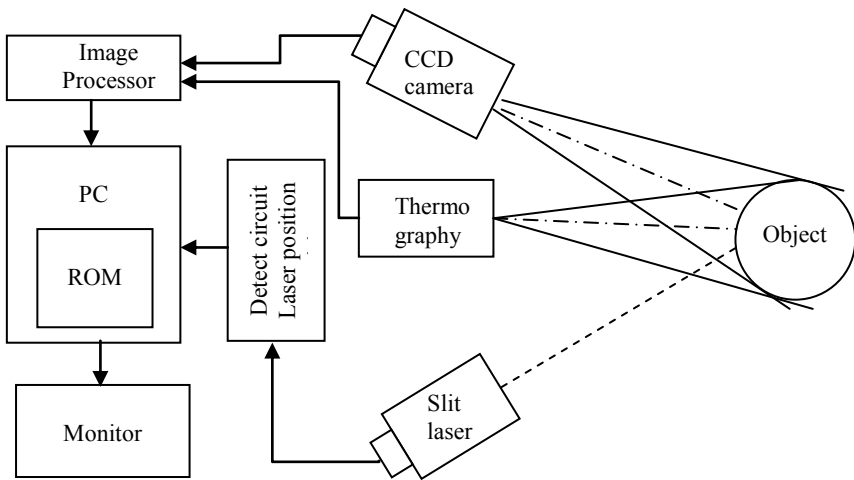
Fig. 1. Measurement system

2   Measurement System

The three-dimensional surface shape measurement system with the temperature information is established using a three-dimensional surface measurement system and a thermography. The measurement system is shown in Fig.1. The three-dimensional surface shape measurement system is composed of a CCD camera and a laser projector [9]-[10], [13]. The three-dimensional surface shape is reconstructed using active triangulation method, and the corresponding temperature data obtained by a thermography is allocated to the reconstructed surfaces of the object on a computer [11]-[12]. The slit laser projector equipped on the Z-axis stage is controlled by computer. The laser streak appeared on the surface of the object is recorded by the CCD camera, and the three-dimensional surface shape measurement of the object is established by analyzing the laser streaks. The thermography records the thermal images simultaneously and the thermal data is allocated to the surface data using the conversion matrices.
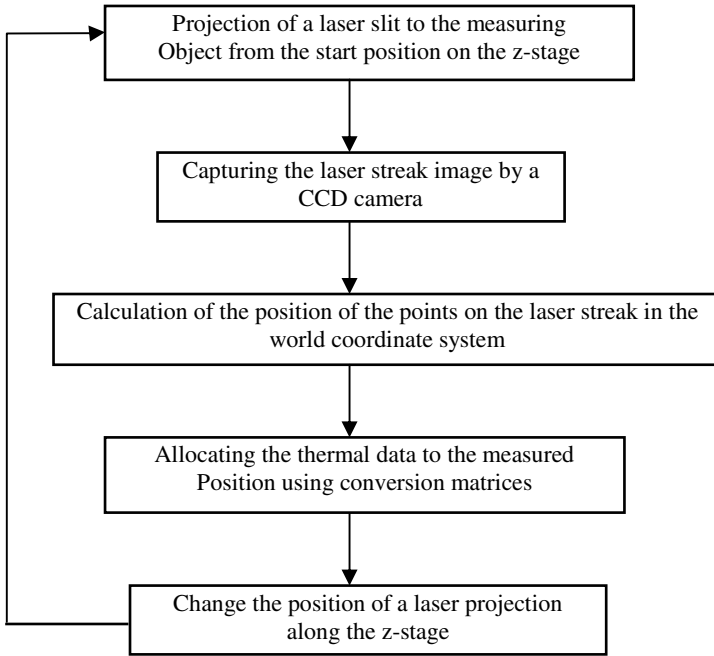
Fig. 2. Flow of the measurement

Fig.2 shows the flow of the measurement. The laser slit ray is projected on the surface of the object. The laser appeared on the surface of the object is captured by the CCD camera. The image of the laser streak is sent to the computer and the computer calculates the position of the points on the laser streak. The image of the object is captured by the thermography simultaneously and the thermal data is allocated to the positional data using a conversion matrices. The position of the laser projector is changed and the above procedure is repeated until the laser reaches at the edge of the object.

3   Measurement Principle

The three-dimensional surface shape is measured using the method of active triangulation. The slit laser projected on the surface of the object is observed by CCD camera and these images are analyzed using image processing technique. The corresponding temperature data obtained by a thermography is allocated to the reconstructed surfaces of the object simultaneously on a computer.

Fig.3 shows the relation between camera coordinate system and world coordinate system. A measurement point is located as the intersection between the laser plane and the straight line from a camera focus.
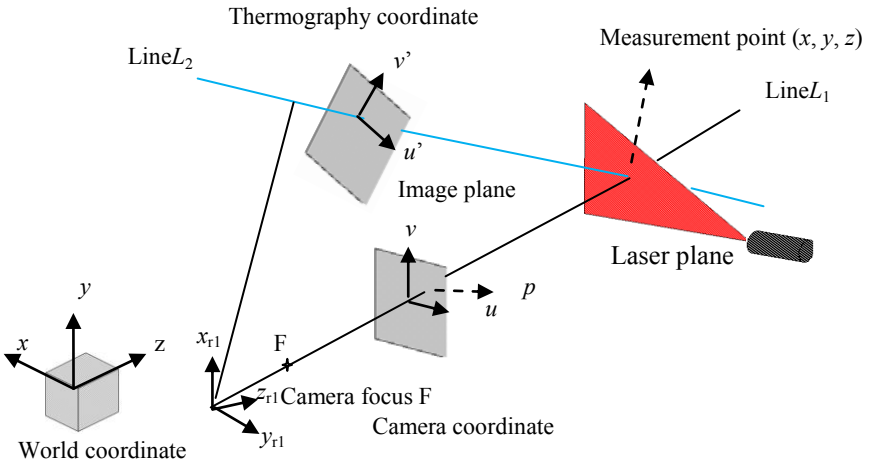
Fig. 3. The relation between camera coordinate system and world coordinate system

3.1   Calibration of a Camera and a Thermography

In order to measure the shape of the object with the temperature, the calibration of the CCD camera and the thermography have to be carried out in advance [15]. A standard cube which has filaments at the each corner is used in order to match the coordinates among the CCD camera coordinate system, the thermography coordinate system and the world coordinate system. Since the filament generates the light and the heat, the image can be recorded by both the CCD and the thermography.

Fig. 4. The calibration setup

```
┌─────────────────────────────────┐
│   Setting the Standard cube with │
│      filaments at the corner     │
└─────────────────────────────────┘
```

```
┌─────────────────────────────┐        ┌─────────────────────────────┐
│   Capturing the image of the │        │   Capturing the image of the │
│  standard cube by the CCD camera │    │  standard cube by the thermography │
└─────────────────────────────┘        └─────────────────────────────┘
```

```
┌─────────────────────────────┐        ┌─────────────────────────────┐
│     Detection of the         │        │   Detection of the position  │
│   position of filaments set  │        │  of filaments set at the corner │
│    at the corner of the cube │        │        of the cube           │
└─────────────────────────────┘        └─────────────────────────────┘
```

```
┌─────────────────────────────┐        ┌─────────────────────────────┐
│   Generation of the relation │        │ Generation of the relation matrix │
│   matrix between the CCD     │        │   between the thermo-image   │
│   coordinates and the world  │        │   coordinates and the world  │
│        coordinates           │        │        coordinates           │
└─────────────────────────────┘        └─────────────────────────────┘
```

```
┌─────────────────────────────────┐
│   Generation of the conversion   │
│  matrices from world coordinates to │
│     thermo-image coordinates     │
└─────────────────────────────────┘
```

Fig. 5. Flow of the calibration procedure

The calibration setup is shown in Fig.4. The standard cube is set along the z-stage direction to be adjusted with the laser slit plane. The image of the standard cube is recorded simultaneously by the CCD camera and the thermography is used to determine the calibration parameters. Fig.5 shows the flow of the calibration procedure of the proposed system. The image of the standard cube with filament marker is captured simultaneously by both the CCD camera and the thermography. The position of the filament marker is detected on CCD image and the thermo image respectively. As the positions of the filament makers are already known in the world coordinate system, the relation matrices between coordinate systems can be generated using the positions of the markers. After determining the relation matrices among the world coordinate system, CCD coordinate system and thermography coordinate system, the conversion matrices from the world coordinate system to the thermo-image coordinates can be generated. The correspondence between three dimensional position of the object and the thermal data can be found using the conversion matrices and it is used for the allocation of the thermal data to the reconstructed shape of the object on the computer.

The relation between the CCD camera coordinates (u, v) and the world coordinates (x, y, z) fixed on the standard cube is as follows.

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} \\ k_{21} & k_{22} & k_{23} & k_{24} \\ k_{31} & k_{32} & k_{33} & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{1}$$

Where k_{11} to k_{33} are parameters that consider the rotation, scale and displacement between camera coordinates and world coordinates. These parameters are determined by feeding some corresponding positions between camera coordinates and world coordinates. In our system, the corners of the cube where the filament marker is set are used for the feeding points. k_{11} to k_{33} are determined by feeding over 6 corresponding points to equation (1). Laser projector moves along the Z-axis by every 1 mm and z coordinate is indicated for the amount of movements of the Z-axis stage. After all, the function for the world coordinates is as follows.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} k_{31}u - k_{11} & k_{32}u - k_{12} \\ k_{31}v - k_{21} & k_{32}v - k_{22} \end{bmatrix}^{-1} \cdot \begin{bmatrix} (k_{14} - u) - (k_{33}u - k_{13})z \\ (k_{24} - v) - (k_{33}v - k_{23})z \end{bmatrix} \tag{2}$$

Where, (u, v) is the CCD camera coordinate. All points on the laser steaks are converted to the world coordinates and the shape of the streak appeared on the surface object is estimated. The whole shape of the object is reconstructed by accumulating of the points in the laser streaks appeared on the surface of the object [14].

The relation between the thermal-image coordinates (u_s, v_s) and the world coordinates(x, y, z) is also as follows.

$$t\begin{bmatrix} u_s \\ v_s \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{3}$$

These parameters can be determined in a same way with the CCD camera one. The data in the thermo-image is allocated to the surface data using equation (3).
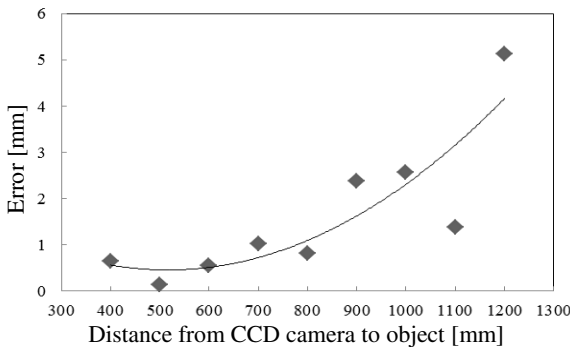
Fig. 6. Measurement accuracy

4   Experiment

4.1   Performance of the System

Since the size of the object appeared in the CCD camera image is varied depend on the distance of the object from the CCD camera and the quantitative measurement is established from images, the distance of the object form the CCD camera influences the measurement accuracy. Therefore, the accuracy of the measurement was evaluated on changing the distance of the object from CCD camera. Fig. 6 shows the measurement accuracy in this experiment. When the distance of the object from CCD camera was not over 800mm, the error was less than 1mm.
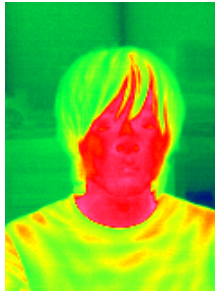
Fig. 7. Thermal image (2D)
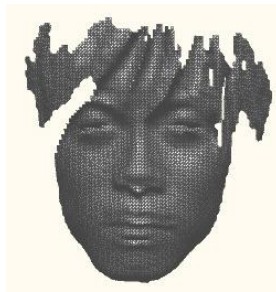
(a) Reconstructed human face on the computer (3D)
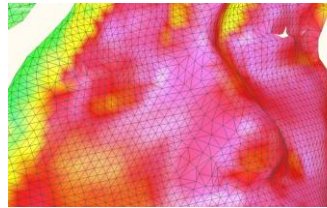
(b)The three-dimensional thermal image   (c) Magnified human face in triangle meshes

Fig. 8. Measurement result

4.2   Result of the Measurement

Thermograph is utilized in various fields, not only engineering fields but also medical care, etc. The measurement result of a human face is introduced for one of the examples of 3D temperature measurement.

Fig.7 shows the two dimensional thermal image and Fig.8 (a) shows 3D reconstructed face using CCD camera and laser projector. The allocated results of thermal data on 3D reconstructed shape are shown in Fig.8 (b) (c). Though Original 2D image has not the quantitative area information such as a size and position of a heat source, the 3D thermal data includes the quantitative position information. The quantitative information can be utilized for the analysis of the heat radiation.

5   Conclusion

The three-dimensional surface temperature measurement system was introduced in this paper. The system measures the 3D shape of the object and the surface temperature of the object is allocated simultaneously on the computer. The obtained data can be used for a quantitative analysis of a heat radiation considering the area and the roughness of the heat source object. Generally, two dimensional images recorded from different angles cannot be connected each other, but the 3D shape data can be connected by adjusting the coordinates on a computer. Therefore, a high resolution model can be reconstructed by connecting the 3D data obtained from different angles.

References

1. Torras, C.: Computer vision: Theory and industrial application. Springer, Heidelberg (1992)
2. Three-dimensional computer vision, A geometric viewpoint. Olivier Faugeras (2006)
3. Ikeda, M.: High Speed 3D Range Capture Techniques, The Institute of Information and Television Engineers. ITE Technical Report 34(16), 9–14 (2010)
4. Shintaku, H., Oike, Y., Ikeda, M., Asada, K.: Design of FPGA for Real-Time 3-D Imager Control and FastData Transmission. Institute of Electronics, Information, and Communication Engineers D-11-6 (2003)
5. Hattori, K., Asada, K., Ikeda, M.: Hardware design for Rea-time 3D Mesh Generation. ITE Technical Report 34(43), 57–60 (2010)
6. Forest, J., Salvi, J.: A review of laser scanning three-dimensional digitisers. In: Proceedings of the 2002 IEEE/RSJ (2002); ISBN:0-7803-7398-7
7. Tsai, R.Y.: A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the Shelf TV Cameras and Lenses. IEEE Journal of Robotics and Automation RA-3(4), 323–344 (1987)
8. Ueshiba, T., Tomita, F.: Calibration of Multi-camera Systems Using Planar Patterns. In: IPSJ SIG17(CVIM8), vol. 44, pp. 89–99 (2003)
9. Nezuka, T., Ikeda, M., Asada, K.: A smart position sensor with row parallel position detection for high speed 3-D measurement. In: ESSCIRC 2002, pp. 101–104 (2002)

10. Yachide, Y., Ikeda, M., Asada, K.: Time-Division-Based Multiple-Viewpoint 3-D Measurement System for Real-Time, High-Speed, and High-Accuracy Model Movie Acquisition. The Institute of Image Information and Television Engineers 62(3) (2008)
11. Lin, H.-Y., Subbarao, M.: A Vision system for Fast 3D Model Reconstruction. In: IEEE CVPR, vol. 2 (2001)
12. Oike, Y., Ikeda, M., Asada, K.: A Pixel-Level Color Image Sensor With Efficient Ambient Light Suppression Using Modulated RGB Flashlight and Application to TOF Range Finding. In: VLSI Circuits Symposium, pp. 298–301 (2004)
13. Oike, Y., Ikeda, M., Asada, K.: 640 × 480 Real-Time Range Finder Using High-Speed Readout Scheme and Column-Parallel Position Detector. In: IEEE Symp. VLSI Circuits Dig of Tech. Papers, pp. 153–156 (2003)
14. Koo, B.-K., Choi, Y.-K., Chen, S.-I.: 3D human whole body construction by contour Triangulation. IEICE Trans. Inf. & Syst. E87-D(1), 233–243 (2004)
15. Levoy, M., et al.: The Digital Michelangelo Project: 3D scanning of large statues. In: SIGGRAPH 2000 Procedings, New Orleans, LA (2000)

A Markov Random Field Model for Image Segmentation Based on Gestalt Laws

Yuan Ren, Huixuan Tang, and Hui Wei

Lab of Algorithm for Cognitive Model, School of Computer Science, Fudan
University, No. 220 Handan Road, Shanghai 201203, P.R. China
{renyuan,0124143,weihui}@fudan.edu.cn

Abstract. This paper proposes a Markov Random Field model for image segmentation based on statistical characteristics of contours. Different from previous approaches, we use Gestalt Laws of Perceptual Organization as natural constraints for segmentation by integrating contour orientations into segmentation labels. The basic framework of our model consists of three modules: foreground/backgraound separation, attentive selection and information integration. This model can be realized for both automatic and semiautomatic image segmentations. Our algorithm achieves smooth segmentation boundaries and outperforms other popular algorithms.

Keywords: Markov Random Field, Image segmentation, Contour, Gestalt Laws of Perceptual Organization, Saliency.

1 Introduction

Image understanding is the ultimate goal for computer vision research. Every digital image is a set of pixels representing colors and these pixels can make sense only when placed together sequentially. A fundamental step for image understanding is to separate objects of interest from others, i.e. foreground/background separation(FBS). FBS is the foundation of multiple visual perceptual abilities. Since FBS raises image representation up to the object level, as for human being, the cerebrum needs much less space to store dynamic visual stimulus [1].

The principal difficulty of FBS lies in the image ambiguity. Since retina images and digitals image can only provide very limited amounts of information, it is difficult to accomplish a seemingly easy task, for example, to understand simple images. Helmholtz has already pointed out the key role of unconscious inference in vision: an image can be successfully analyzed and interpreted only when image information and related prior knowledge are both utilized [2].

A important sort of prior knowledge for FBS is known as Gestalt Laws for Perceptual Organization (GLPO). GLPO suggests that the cerebrum tends to interpret perceived information as simple whole forms and the whole is greater than the sum of its parts, which was initially brought forward by Psychologist

Ehrenfels [3]. Neurophysiologic experiments have revealed that FBS includes three major steps [4]: image edge detection; global information integration and pre-attentive foreground selection. Obviously, GLPO functions mainly in the middle stage. Nevertheless, for lack of computational model, there has been no answer to the question "how is the whole-form realized in the cerebrum which is a parallel distributed organization?", which limits GLPO research to qualitative analysis only. Malik and several fellows in Berkeley studied GLPO based learning algorithms and applied them to edge detection [5,6] and contour completion [7]. Elder and several fellows in York University studied the ecological statistics of GLPO of contours using similar approaches [8]. Both Elder's and Malik's models are based on Canny edge detector, therefore they can only show the saliency of each line segment constructed by Canny edges but yet cannot solve problems such as missing or inaccurate Canny segments. Although Malik, Elder and many other researchers have made outstanding contributions to establishing global optimized computational models, few is biologically feasible. Sokolov et al. proposed two hypotheses on the neural basis of GLPO [9] arguing that GLPO may result from either the hierarchical organization of visual system or the synchronous vibration of neural dynamic system. In recent years, Ehrenstein and several neurobiologists have inclined to believe that GLPO results from the hierarchically organization of the visual receptive field. [10].

In this paper, we establish a mathematical framework of FBS mechanism for visual cortical areas V1 and V2. The current work mainly focuses on three aspects: (1) We explore how FBS can be accomplished in the middle layer through basic neural computations. The computing procedure should be largely consistent with the neural computational mechanisms and should be extendible (2) We propose a computational model in accordance with physiologic experiment for FBS. The model should be able to optimize global information via local computing. (3) By quantitatively describing GLPO in the probability space, we fully integrate the underlying constraints into the design of our MRF model.

The remainder of this paper is organized as follows: Section 2 introduces the design of our MRF model and the solution; Section 3 specifically discusses the determinations of the objective functions of the MRF by quantitatively analyzing GLPO in probability space; Section 4 illustrates a group of experimental results obtained by automatic segmentation algorithm and that by semiautomatic algorithm; and finally Section 5 concludes this paper.

2   The MRF Model and Its Solution

Since FBS is a strongly subjective problem, in recent years, there have been an increasing number of image segmentation algorithms aided by manual interventions. An manual-intervention-aided algorithm requires a user either to initially label a few sample points of an image [11] or to segment image pairs at one time [12]. This kind of design approaches provides a feasible solution for image segmentation tasks. In the current work, we use two approaches to guide the

extraction of a foreground: (1) The attentive selection based automatic FBS uses the spectral residual approach [13]. The automatic algorithm first computes the saliency map of an image and then determines the segmentation through the inferential process of our MRF model. (2) The semi-automatic FBS uses user-stroke. First a user draws strokes on the image to decide the labels (foreground or background) of certain pixels; then the segmentation is also generated through the inferential process of our MRF model.

Each of the two approaches corresponds to a distinct realization of attentive selection which is a function of V4. The automatic algorithm determines the positions of objects in the foreground by locating high frequent signals while the semi-automatic one by letting the user decide partially. Suppose that the attention saliency of V4 is spatially independent and the two hypotheses just mentioned about foreground saliency are true, then a abstract FBS task turns to such a mathematical problem: given an image $I(x, y)$ on grid Ω and the probabilities of pixels being labeled as foreground, the algorithm should determine the label of each grid in Ω and find the set $R \subset \Omega$ of the image foreground. Naturally, there are three requirements for foreground and background labels: (1) The probability $P((x, y) \in R)$ of a pixel being labeled as foreground should be consistent with the constraint condition or the foreground saliency determined by the user stroke. (2) The location of the contour curve ∂ of R should be where the grayscales or the textures changes greatly. (3) ∂ should accord with the statistical properties of natural contours, such as closure, continuity, smoothness and etc.

(a) 10 kinds of labels   (b) A possible segmentation
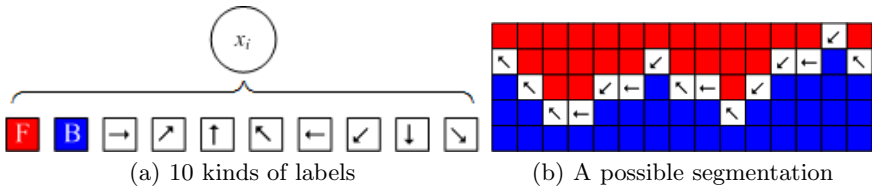
Fig. 1. Label definitions and an example of label distribution. The contour labels are marked by the corresponding directions. The foreground label is marked by red color and the background label by blue color.

Equally, associate every pixel with a label $L(x, y)$ defined on Ω. Unlike previous work, we define 10 labels (Fig.1). Each pixel may not just be specified as foreground or background but as contour. Contour labels are defined by 8 directions with equal intervals. Given a contour curve, the label of each point is determined by the clockwise tangent direction of the curve at that point. As a representation of the segmentation, it is redundant to label the contour with its local directions, nevertheless, that helps to encode GLPO into the corresponding objective function of the MRF.

An MRF is a set of random variables with spatial correlations. Hammersley and Clifford have proven that the probability density of an MRF can be written as the product of several marginal probabilities [14]

$$P\left(X\right) = \prod_{C} P\left(X_C\right) \tag{1}$$

where X_C is the set of the random variables in the Cth clique of the field(a graph). In order to simplify the MRF, we only consider low-order statistical correlations, i.e. the relation between two variables, then (1) can be rewritten as

$$P\left(X\right) = \prod_{i} P\left(x_i\right) \prod_{i,j} P\left(x_i, x_j\right) \tag{2}$$

where x_i denotes the ith variable in the field. Let $\varPhi\left(X\right) = -\log_2 P\left(X|S, K\right)$, $\phi\left(x_i\right) = -\log_2 P\left(x_i|s_i\right)$ and $\varphi\left(x_i, y_i\right) = -\log_2 P\left(x_i, y_i|\kappa_i, \kappa_j\right)$. Then we have

$$\varPhi\left(X\right) = \sum_{i} \phi\left(x_i\right) + \sum_{i,j} \varphi\left(x_i, x_j\right) \tag{3}$$

The data cost $\sum_{i} \phi\left(x_i\right)$ measures the data-driven parts in the inferential process and attention-controlled part in the FBS model. The smoothness cost $\sum_{i,j} \phi\left(x_i, x_j\right)$ measures the fitting degree between the values (the labels) of two variables and the prior knowledge and reflects the statistical laws driven by GLPO in the FBS model.

Fig.2 shows the structure of our network model. The primary inferential structure consists of variables and the connections between them. The solid circles represent the labels assigned to the variables(pixels) and subject constraint from the foreground saliency of V4, namely the data cost. The solid squares between solid circles represent the smoothness cost controlled by GLPO. It is worth noting that GLPO are not completely independent of the input images, e.g. from the discussion coming in Section 3 it can be seen that the law of similarity is driven by edge saliency as well as other features.

There are three sorts of methods for solving discrete MRF: Iterative Conditional Mode, Message Passing and Binary Optimization. Among them, the latter two usually perform better in optimizations for visual applications. As for our model, Message Passing is more preferable since its computational mechanism
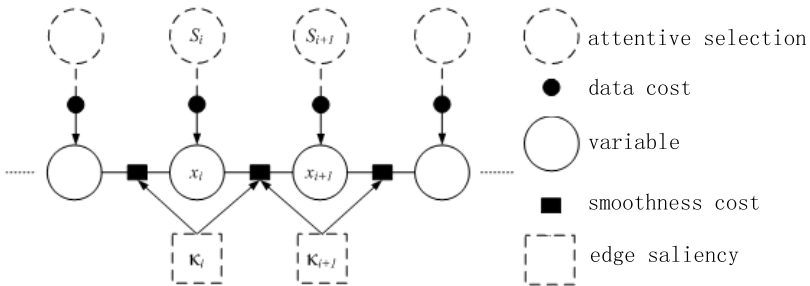
Fig. 2. The MRF model for image segmentation in this paper

is in accordance with the neural computational mechanism [15,16]; more importantly, when there exist multiple solutions for one image Message Passing can usually find all of them while Binary Optimal can find only one. As a Message Passing based method, the tree-reweighted belief propagation algorithm [17] iteratively updates the information that each node will send to every one of its neighboring nodes by

$$m_{s,t}(k) \leftarrow \min_{j \in L} \left\{ \phi_s(j) + \sum_{u \in N_s, u \neq t} m_{u,s}(j) + \varphi_{s,t}(j,k) \right\} \tag{4}$$

where $\phi_s(j)$ denotes the data cost of node s being labeled as j and $\varphi_{s,t}(j,k)$ denotes the smoothness cost of a pair of nodes s and t being labeled as j and k respectively. When the iteration converges, compute the message that each node receive:

$$b_s(k) \leftarrow \sum_{t \in N_s} m_{t,s}(k) \tag{5}$$

where N_s denotes the set of all neighboring nodes of s. Finally all the labels are given by

$$l_s = \min_{k \in L} b_s(k) \tag{6}$$

In order to make each node affect the solutions of other nodes as much as possible, the variables are sorted according to the uncertainties of their solution selections. The higher uncertainty, the later update and the more information. In our MRF model, the uncertainty is measured by

$$UC_i = \frac{\sum_k b_s(k) \exp(-b_i(k))}{\sum_k \exp(-b_i(k))} + \log_2 \sum_k \exp(-b_i(k)) \tag{7}$$

which has a form similar to information-entropy.

The experiments reveal that the pixels closer to the user stroke subject to more constraints; the corresponding uncertainties are lower and the information is therefore earlier updated. The solutions can be selected more easily for pixels on clearer edges and the information is updated relatively earlier. On the contrary, at corners and discontinuous positions, ambiguities are higher and labels cannot be determined until the algorithm converged. This maximum-constraint-first mechanism concentrates the more computational resources to the most crucial variables and labels and consequently improves the validity and proficiency of the FBS. In the following section, we discuss the quantitative descriptions of GLPO and utilize them to explicitly determine the cost functions which defines our MRF model.

3   The Cost Functions and the Quantitative Analysis of GLPO

In this paper, we define the data cost by

$$\phi(l_i) = -\log_2 P(l_i|s_j) \tag{8}$$

where s_i denotes the saliency of a pixel. According to Bayes's formula, the conditional probability $P(l_i|s_j)$ is given by

$$P(l_i|s_j) = \frac{P(s_j|l_i)\,P(l_i)}{\sum\limits_k P(s_j|l_k)\,P(l_k)} \tag{9}$$

The prior probability $P(l_i)$ can be obtained through non-parametric statistics and $P(s_j|l_i)$ from exponential regression estimations. The saliency map of an image is computed by the algorithm proposed in [13].

The smoothness cost can be defined in multiple ways according to different needs. Considering the position proximity between a pair of pixels, we first define smoothness cost by the joint distribution of label given the neighborhood

$$\varphi(l_i, l_j) = -\log_2 P(l_i, l_j|N_{x,y}) \tag{10}$$

where $N_{x,y}$ denotes the union of the neighborhoods of the two pixels. For GLPO, the law of continuity holds that points connected by straight lines or curves are seen as following the smoothest path. The law of closure holds that things are grouped together if they seem to complete certain entity. In our model, the closure of a foreground contour is determined by the labels of the foreground and the background: the pixel at the direction indicated by the label of a pixel on the contour is usually on the contour too; and the direction also determines the labels of the pixels beside.

We define co-occurrence matrixes of label to collect statistics from the Berkeley image data set and use the obtained matrixes to determine $P(l_i, l_j|N_{x,y})$. It is observed in the experiments that for almost every (x, y), $P(l_i, l_j|N_{x,y})$ concentrates at continuous directions, which well reflects the law of continuity. Kovács et al have pointed that this constraint can greatly increase the robustness of FGS [18].

The saliency proximity is also a reasonable consideration for our model. The law of similarity holds that similar items tend to be grouped together. Accordingly, we define another form of smoothness cost by the joint distribution of the label given the saliency

$$\varphi(l_i, l_j) = -\log_2 P(l_i, l_j|s_m, s_n) \tag{11}$$

According to Beyes's formular, we have

$$P(l_i, l_j|\kappa_m, \kappa_n) = \frac{P(\kappa_m, \kappa_n|l_i, l_j)\,P(l_i, l_j)}{\sum\limits_u \sum\limits_v P(\kappa_m, \kappa_n|l_u, l_v)\,P(l_u, l_v)} \tag{12}$$

where κ denotes the Gaussian curvature of $I(x, y)$ at a point. $P(l_i, l_j)$ can also be obtained through non-parametric statistic and $P(s_m, s_n|l_i, l_j)$ can be approximated by scale mixtures of Gaussians. More details about this algorithm can be found in literature [19].

It is worth noting that even though different laws in GLPO and prior conditions determine distinct characteristics of the label distribution, all the

probabilities defined above will superimpose in a linear way when substituted into (3). This is highly consistent with the conclusion that individual laws of GLPO are independent.

4   Experimental Results

All Tests of our algorithm were performed on the Berkeley image segmentation data which provides 200 images with approximately 1000 labels each. We simulate the function of V1 using the edge detection algorithm proposed in [5]; model the pre-attentive selection of V4 by spectral residual for automatic segmentation and by user stroke for semiautomatic segmentation.

First we compare our automatic segmentation with Normalized Cuts which is a multi-scale spectral segmentation algorithm [20]. The comparative results are illustrated in Fig.3. Obviously, our algorithm does not just generate smoother segmentation boundaries but highlight the foregrounds of images with complex textures more clearly. This should be attributed to the fact that spectral segmentation based algorithms can only classify pixels into clusters rather than picking foregrounds out of the clusters. What's more, the control of spectral segmentation algorithm over shapes is limited to the topological structure of an MRF model, nevertheless the statistical properties of contours ensured by such structure are somehow different from those of natural contours. Our approach
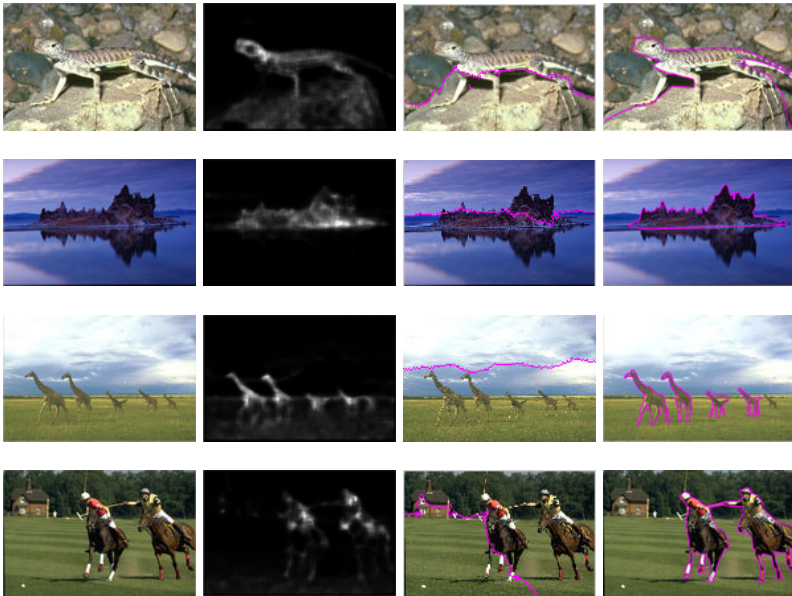
Fig. 3. Automatic FBS. From the left to the right are original images, the significant patterns, Normalized Cuts segmentations and ours.

makes the foreground contours more natural through modeling the statistical properties of contours according to GLPO.

We also compare our algorithm with several others for semiautomatic segmentation. We design a simple interactive system to receive a user's strokes. When the system starts, it shows several images to the user. After the user outlines the foreground/background, the system will automatically perform edge detections and FBS and then display the results. The user may add, remove or modify a stroke according to the current result for possible improvement. The final segmentations obtained with our approach and those with the Grabcut [21]+Lazy Snapping [11] are shown in Fig.4. Since our approach optimizes marginal smoothness, it generates smoother segmentation boundaries.
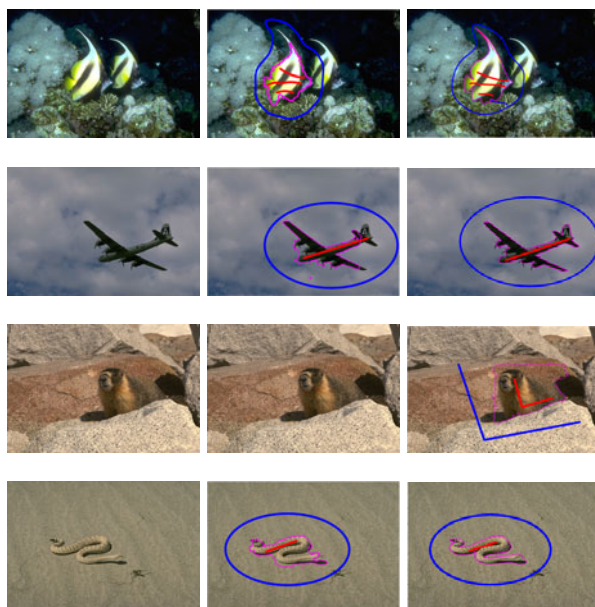
Fig. 4. Semiautomatic FBS. From the left to the right are original images, Grabcut+Lazy Snapping based segmentations and ours.

5   Conclusions

In this paper we propose a FBS model based on MRF and GLPO. Compared with previous segmentation algorithms, our model is more robust and results in smoother segmentations. Related neurophysiologic discoveries has verified that the segmentation mechanism in the visual cortex is guided by edge detection, indicating that the current approach is more deeply rooted in physiology. By quantitatively analyzing GLPO and integrating the corresponding constraints into the definition of our MRF, we improve the segmentation mechanism with the natural statistical properties of contours.

Unlike most other segmentation algorithms, the principal purpose of the current study is to establish a physiologically and computationally feasible model as the foundation for future research on neural vision, machine vision and cognition. Since our algorithm has a practical significance for engineering computation, it is necessary to further analyze the connections and differences between other algorithms and ours.

(1)The computing processes of spectral segmentation based approaches such as [20] differ much from that of our approach. In terms of computational theory, the spectral approaches consider more about the distances between neighbors or pixels but little about the features of contours. This is the main reason why they cannot achieve smooth segmentations sometimes. Moreover, a wrong estimation of the number of clusters often leads to undesirable segmentations, which explains why spectral algorithms cause over-segmentations occasionally. Our algorithm is guided by a user's strokes or the saliency map of an image, the final segmentation is therefore more natural to the eyes.

(2)The MRF model based on Scale Mixtures of Gaussians has a structure similar to that of our model. Like spectral segmentation, MRF based approaches do not consider the effects of edges, therefore previous MRF models can hardly be combined with GLPO. This also accounts for the rough boundaries obtained with Gaussian mixture model based approaches.

(3)Learning based segmentation approaches have explored the statistical properties of GLPO earliest. Nevertheless, since most of them carry out the inferential process of GLPO on the basis of filtering pseudo edges only, this sort of algorithm yet cannot solve the problems caused by edge detectors, such as undetected edges, inaccurate localizations and other sort of errors.

Neuroscientists tend to believe that there exists a fast color segmentation system guiding FBS tasks [22]. One of the system's main functions is to make the MRF model to concentrate computational resources on more valuable part. Another function is to increase the information bandwidth of MRF model and the computing speed with subpixel-level presentations of images. The design and the realization of this system will be explored in our future work.

References

1. Mazza, V., Turatto, M., Umilt, C.: Foregroundcbackground segmentation and attention: A change blindness study. Psychological Research 69, 201–210 (2005)
2. von Helmholtz, H.: Handbuch der physiologischen Optik. Handbook of Physiological Optics. Leopold Voss (1867)
3. Ehrenfels, C.V.: On gestalt-qualities. Psychological Review 44(6), 521–524 (1937)
4. von der Heydt, R., Peterhans, E.: Mechanisms of contour perception in monkey visual cortex. i. lines of pattern discontinuity. The Journal of Neuroscience 9(5), 1731–1748 (1989)

5. Martin, D.R., Fowlkes, C.C., Malik, J.: Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Trans. Pattern Anal. Mach. Intell. 26, 530–549 (2004)
6. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings of Eighth IEEE International Conference on Computer Vision, ICCV 2001, vol. 2, pp. 416–423 (2001)
7. Ren, X., Malik, J.: A probabilistic Multi-scale Model for Contour Completion Based on Image Statistics. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2350, pp. 312–327. Springer, Heidelberg (2002)
8. Elder, J.H., Goldberg, R.M.: Ecological statistics of gestalt laws for the perceptual organization of contours. Journal of Vision 2(4) (2002)
9. Sokolov, E.: The problem of gestalt in neurobiology. Neuroscience and Behavioral Physiology 27, 323–332 (1997)
10. Ehrenstein, W., Spillmann, L., Sarris, V.: Gestalt issues in modern neuroscience. Axiomathes 13, 433–458 (2003)
11. Li, Y., Sun, J., Tang, C.-K., Shum, H.-Y.: Lazy snapping. ACM Trans. Graph. 23, 303–308 (2004)
12. Rother, C., Minka, T., Blake, A., Kolmogorov, V.: Cosegmentation of Image Pairs by Histogram Matching - Incorporating a Global Constraint into MRFs. In: International Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 993–1000 (2006)
13. Hou, X., Zhang, L.: Saliency detection: A spectral residual approach. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007)
14. Hammersley, J.M., Clifford, P.: Markov field on finite graphs and lattices (1971)
15. Rao, R.P.N.: Neural Models of Bayesian Belief Propagation, 1st edn., pp. 235–264. The MIT Press (2007)
16. Popple, A.V.: Context effects on texture border localization bias. Vision Research 43(7), 739–743 (2003)
17. Wainwright, M.J., Jaakkola, T.S., Willsky, A.S.: Tree-reweighted belief propagation algorithms and approximate ML estimation by pseudo-moment matching. In: Workshop on Artificial Intelligence and Statistics (2003)
18. Kovács, I., Julesz, B.: A closed curve is much more than an incomplete one: effect of closure in figure-ground segmentation. Proceedings of the National Academy of Sciences 90(16), 7495–7497 (1993)
19. Wainwright, M.J., Simoncelli, E.P.: Scale mixtures of gaussians and the statistics of natural images. In: Advances in Neural Information Processing Systems, vol. 12, pp. 855–861. MIT Press, Cambridge (2000)
20. Shi, J., Malik, J.: Normalized cuts and image segmentation. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 731–737 (June 1997)
21. Rother, C., Kolmogorov, V., Blake, A.: "GrabCut": interactive foreground extraction using iterated graph cuts. ACM Transactions on Graphics 23(3), 309–314 (2004)
22. Møller, P., Hurlbert, A.C.: Psychophysical evidence for fast region-based segmentation processes in motion and color. Proceedings of the National Academy of Sciences 93(14), 7421–7426 (1996)

Weber's Law Based Center-Surround Hypothesis for Bottom-Up Saliency Detection

Lili Lin[1], Wenhui Zhou[2,⋆], and Hua Zhang[2]

[1] College of Information and Electronic Engineering,
Zhejiang Gongshang University, Hangzhou 310018, China
[2] College of Computer Science and Technology,
Hangzhou Dianzi University, Hangzhou 310018, China

Abstract. Inspired by Weber's Law and the biological model of synergistic center-surround receptive field, this paper proposes a center-surround hypothesis for saliency detection. Specifically, this detector defines two types of salient stimuli. One type is local stimulus represented as a set of differential excitation of gradient orientation for each pixel. The other type is global stimulus, which is the relative intensity differences of center region against the overall mean. Then a center-surround model with ring topology structure is designed to extract salient responses of these two types of stimuli. For a given color image, these salient responses are computed on each color channel separately, and then combined linearly to get the final saliency map. Comparison experiments demonstrate this detector not only can generate high quality saliency maps with the same resolution as the input image, but also has stronger response in activation regions and better inhibition performance in other regions.

Keywords: Visual salience, Center-surround hypothesis, Weber's law, Local stimulus, Global stimulus.

1 Introduction

Saliency mechanism plays an important role in visual selective attention. It may provide a rapid and effective strategy to reduce the computational complexity of visual processing. Most bio-inspired saliency models rely on bottom-up or top-down processing. The former is based on the feature integration theory and the center-surround hypothesis. The representative works are those by L. Itti et al. [1]. The latter is a goal-directed saliency analysis and requires the prior knowledge of the task [2]. Alternative integrated models of top-down and bottom-up attention have also been proposed [3]. However, these models have shortcomings of low resolution, poorly defined object boundaries, and expensive computation.

Recently, some simple and effective saliency models are proposed. R. Achanta et al. presented a fast salient region detection method based on low-level features of luminance and color [4]. R. Achanta et al. also introduced a higher precision salient region detector based on frequency-tuned [5]. V. Gopalakrishnan et al.

⋆ Corresponding author.

present a color and orientation distribution based salient region detection framework [6]. In frequency domain, Q. Zhang et al. integrated local saliency, global saliency and rarity saliency into a framework [7]. D. Gao et al. proposed a discriminant saliency detector [8]. These models are usually pure computational ones, and are inspired by the biological concept of center-surround contrast, but are not based on any biological model.

We introduces a novel center-surround hypothesis for bottom-up saliency detection. It is inspired by a biological model and Weber's Law in psychophysics. Compared with other saliency models, our main contributions are followed:

a) We define two types of visual salient stimuli followed Weber's Law. One type is differential excitation of gradient orientation, which represents the property of local stimuli. The other type is relative intensity differences of center region against the overall mean, and it shows the property of global stimulus.

b) Inspired by a center-surround pattern of biological vision, we design a center-surround architecture which has ring topology structure composed of a center cell and its surround cells. Moreover, we extract center-surround interactions from salient stimulus image rather than original image.

c) In order to obtain the final visual saliency map, we propose a linear combination strategy to integrate all salient stimulus responses of each color channels.

2 Weber's Law Based Visual Salient Stimuli

2.1 Local Salient Stimulus

Based on the Weber's Law and the fact that human vision is sensitive to the gradient magnitude and orientation, we define a local salient stimulus represented as a differential excitation vector of gradient orientation for each pixel.

Let $I(u_0, v_0)$ be the intensity of current pixel (u_0, v_0), then the differential excitation of horizontal gradient $\varepsilon_h(u_0, v_0)$ can be formulated as followed.

$$\varepsilon_h(u_0, v_0) = \begin{cases} \frac{|I(u_0+1,v_0)-I(u_0,v_0)|}{I(u_0,v_0)} & u_0 = 0 \\ \frac{|I(u_0+1,v_0)-I(u_0-1,v_0)|}{I(u_0,v_0)} & 0 < u_0 < width - 1 \\ \frac{|I(u_0,v_0)-I(u_0-1,v_0)|}{I(u_0,v_0)} & u_0 = width - 1 \end{cases} \tag{1}$$

A similar computation is used for the differential excitation of vertical gradient $\varepsilon_v(u_0, v_0)$. Then, differential excitation of any gradient orientation $\varepsilon_\theta(u_0, v_0)$ can be formulated as

$$\varepsilon_\theta(u_0, v_0) = \varepsilon_v(u_0, v_0) \cdot \sin(\theta) + \varepsilon_h(u_0, v_0) \cdot \cos(\theta) \tag{2}$$

Let K be the number of bins in gradient orientation histogram. The local salient stimulus $\boldsymbol{s}_{local}(u_0, v_0)$ of current pixel can be defined as a differential excitation vector of its quantized gradient orientation, i.e.

$$\boldsymbol{s}_{local}(u_0, v_0) = [\varepsilon_{\theta_0}(u_0, v_0), \varepsilon_{\theta_1}(u_0, v_0), \cdots, \varepsilon_{\theta_K}(u_0, v_0)]^T \tag{3}$$

Therefore, the local stimulus images can be constructed by local salient stimuli of all pixels.

$$\mathbf{s}_{local} = \left[\boldsymbol{\varepsilon}_{\theta_0}, \boldsymbol{\varepsilon}_{\theta_1}, \cdots, \boldsymbol{\varepsilon}_{\theta_K}\right]^T \tag{4}$$

where $\boldsymbol{\varepsilon}_{\theta_k}$ is differential excitation vector of all pixels with gradient orientation θ_k.

(a) biological model   (b) computational model
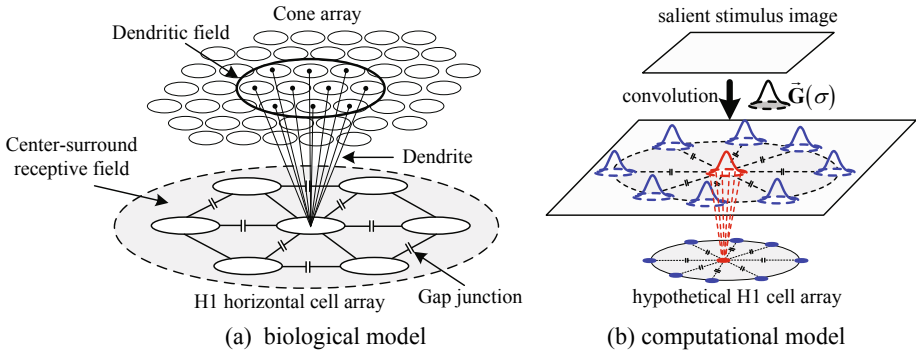
Fig. 1. Simplified compartmental model and the proposed computational model

2.2 Global Salient Stimulus

Global salient stimulus is determined by the global contrast between center region and image background. In our works, the whole image mean is considered as image background, and the global saliency stimulus is defined as the intensity distance between the center region of center-surround architecture and the whole image mean. So the global saliency stimulus of current pixel (u_0, v_0) can be formulated as followed.

$$S_{global}(u_0, v_0) = \left\| g(u_0, v_0, \sigma) - \mu^2 \right\|_2 \tag{5}$$

where μ is the whole image mean, $g(u_0, v_0, \sigma)$ represents the center region with Gaussian scale σ. More details can be referred to the section 3.

3 The Proposed Center-Surround Hypothesis

3.1 Topology Structure of Center-Surround Profile

Here we are mainly inspired by two research results in cognitive neuroscience. The first is the compartmental model of the cone-H1 cell network, which had been used to simulate the synergistic center-surround receptive field of monkey H1 horizontal cells [9]. A simplified compartmental model is shown in Fig. 1(a). The second is the resolution hypothesis in visual attention [10]. The experimental evidences suggest that attention mechanisms can actively enhance the spatial resolution at the attended location. Moreover, the attention dynamics can be

demonstrated by a processing of multiple spatial resolutions with visual search of hierarchical patterns.

Based on the above two neuropsychological evidences, we design a center-surround computational model with ring topology structure by extending the biological compartmental model to the salient stimulus images. The illustration of the ring topology structure of proposed center-surround profile is shown in Fig.1(b). The pixels of salient stimulus image can be regarded as the cone array in Fig.1(a). The hypothetical H1 cell with circular profile can be generated by convoluted with a Gaussian function. Gaussian scale σ determines the spatial resolution of center and surrounds regions.

3.2   Center-Surround Hypothesis and Interaction

According to the center-surround mechanism, we suppose there have statistically significant responses of differential excitation when center-surround organization is on or near the salient location. Similar statistical suppositions have been discussed in many recent literatures [2,4]. Our framework of center-surround hypothesis and interaction are shown in Fig.2.

Assume there are N hypothetical H1 cells (small blue solid circular points) on the surrounds. $l_n(u_0, v_0, r)$ is the location of the nth hypothetical H1 cell on the surrounds circle with radius r. The convolution result $\mathbf{g}_{local}$ between local stimulus images $\mathbf{s}_{local}$ and Gaussian function $\mathbf{G}(\sigma)$ can be written as

$$\mathbf{g}_{local}(\sigma) = \mathbf{s}_{local} * \mathbf{G}(\sigma) = [\varepsilon_{\theta_0}(\sigma), \cdots, \varepsilon_{\theta_K}(\sigma)]^T \tag{6}$$

$\varepsilon_{\theta_k}(l_n(u_0, v_0, r), \sigma)$ is the value of the kth bin in histogram of the nth surround cell. For simplicity, we denote $\varepsilon_{\theta_k}(l_n(u_0, v_0, r), \sigma)$ as $\varepsilon_{\theta_k}(l_n)$. The center-surround interaction of current pixel (u_0, v_0) is

$$\mathbf{L}(u_0, v_0) = [\eta_1^T, \eta_2^T, \cdots, \eta_K^T]^T, \quad where \quad \eta_k = \begin{bmatrix} \varepsilon_{\theta_k}(u_0, v_0,), \\ \varepsilon_{\theta_k}(l_1), \cdots, \varepsilon_{\theta_k}(l_N) \end{bmatrix}^T \tag{7}$$

We propose the local saliency decision by variance analysis of gradient orientation histogram. Let ν_k be the variance of vector η_k. The local saliency decision $S_{local}(u_0, v_0)$ can be defined as a linear combination of all ν_k.

$$S_{local}(u_0, v_0) = \frac{1}{K} \sum_k v_k, \quad and \quad \begin{cases} v_k = \sum_{1 \leq n \leq N} (\varepsilon_{\theta_k}(l_n) - \mu_k)^2 \, p(\sigma_n) \\ \mu_k = \sum_{1 \leq n \leq N} \varepsilon_{\theta_k}(l_n) \, p(\sigma_n) \end{cases} \tag{8}$$

where μ_k is the mean the vector η_k. $p(\sigma_n)$ is the probability of the nth cell.

$$p(\sigma_n) = \frac{\sum_{1 \leq k \leq K} \varepsilon_{\theta_k}(l_n)}{\sum_{1 \leq n \leq N} \sum_{1 \leq k \leq K} \varepsilon_{\theta_k}(l_n)} \tag{9}$$
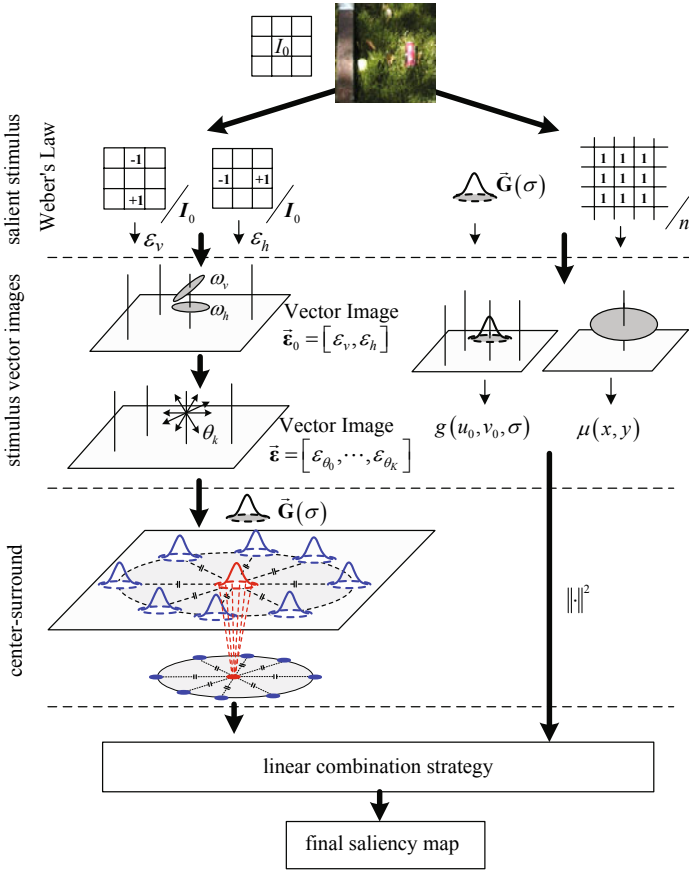
Fig. 2. Framework of center-surround hypothesis and interaction

4   Linear Combination Strategy

4.1   Framework of our Saliency Decision

The framework of our complete saliency decision algorithm is shown in Fig. 3. Firstly, low-level color features are extracted. Since perceptual differences in CIELAB color space are approximately Euclidian [4,11], we use the L*a*b color features. Then we compute the responses of local and global salient stimuli of each color channel, respectively. Local saliency represents the difference between a region and its surroundings. However, it is not sufficient to make decision with only local saliency, because high local saliency values may lie in some global texture regions. Global saliency can not only provide global constrains, but also reduce the effect of background. Finally, we generate the final saliency map by a linear integration strategy of local and global saliency.

4.2 Linear Combination Strategy

For CIELAB color space, the global saliency decision vector and the local saliency decision vector of pixel (u_0, v_0) can be formulated as followed.

$$\begin{cases} \mathbf{S}_{global}(u_0, v_0) = \left(S^l_{global}(u_0, v_0), S^a_{global}(u_0, v_0), S^b_{global}(u_0, v_0) \right)^T \\ \mathbf{S}_{local}(u_0, v_0) = \left(S^l_{local}(u_0, v_0), S^a_{local}(u_0, v_0), S^b_{local}(u_0, v_0) \right)^T \end{cases} \quad (10)$$

The idea of our integration strategy of local and global saliency is to enhance the response of local contrast, and to inhibit the response of background. Therefore, we consider the global saliency as global constrains for local saliency. Then the weighted local saliencies of different color features are linearly combined. The final saliency map can be expressed as

$$S_{final}(u_0, v_0) = \mathbf{S}^T_{global}(u_0, v_0) \cdot \mathbf{S}_{local}(u_0, v_0) \quad (11)$$

In fact, our combining strategy is a soft-decision integration strategy. That is, we do not perform binary threshold operation during the integration process, as shown in Fig. 3. As anticipated, the local saliencies have strong responses at some global edges. The global saliencies can weaken the responses of global texture regions. Therefore, the saliency weighted by global saliency can effectively inhibit the effects of global texture and background. It should be noted that the salient maps shown in Fig. 3 are not actual values in whole saliency calculation process, and they are normalized to [0,255] only for show.
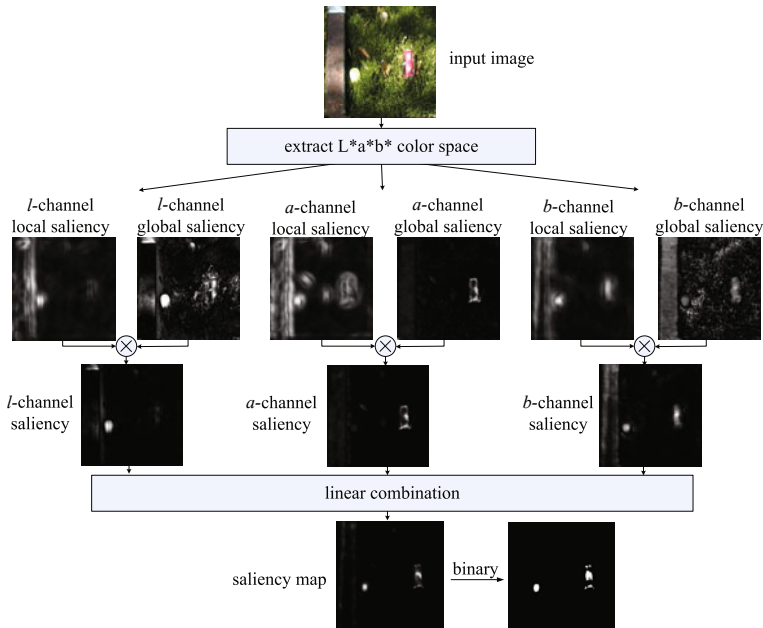
Fig. 3. Framework of complete saliency detection algorithm

5   Experiments and Discussion

The experiments are performed on images from iLab image database at USC[1] and from MSRA Salient Object Database [12]. The saliency detection results of our method are compared with those of Itti's model [1], R. Achanta's global saliency detector [5] and E. Rahtu's salient region detector [13]. The saliency maps of Itti's model are generated using Great Matlab Saliency Toolbox[2]. Achanta's global results are produced using the their program[3]. The binary saliency regions of Rahtu's are generated using his program[4]. The parameters of our method are $N = 8$, $K=8$, and the binary threshold is set to 0.1 with the saliency value range of [0, 1]. Comparison results are shown in Fig.4.

Obviously, Itti's model can only give the rough location of salient regions and cannot extract exact boundaries of salient regions. Achanta's and Rahtu's methods can get salient regions, but are very sensitive to the complex background with high contrast, even may extract incorrect salient regions in some cases. This can be seen especially from their results in the 1st, 4th, 6th, 9th columns. Since combining local and global saliency decisions, our method can extract

Original images

Itti's saliency maps

Achanta's global saliency maps

Rahtu's binary salient maps

Our saliency maps

Our binary saliency maps
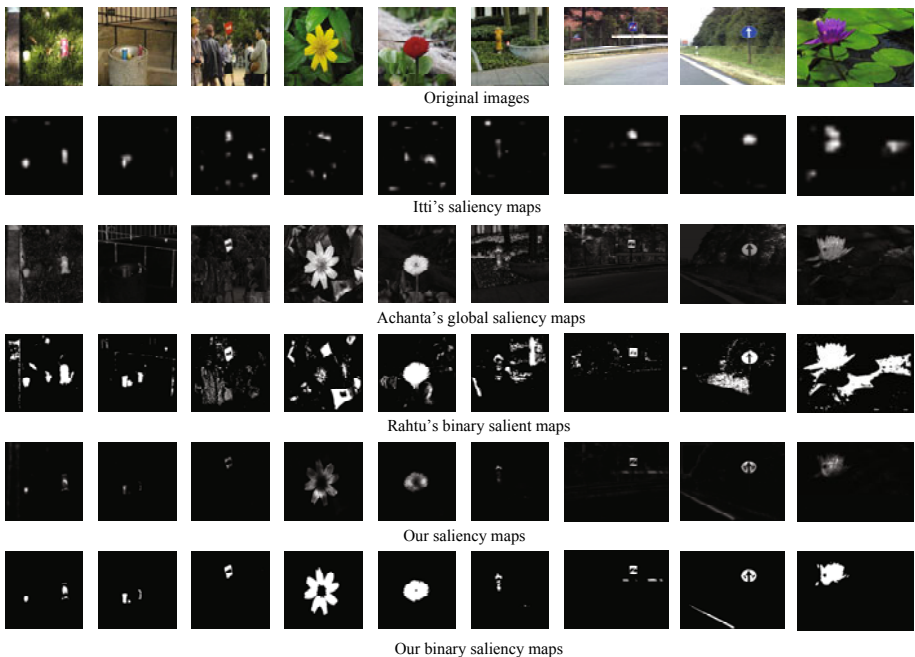
Fig. 4. Comparison results on images from iLab and MSRA Database

more exact salient regions and preferably inhibit the saliency values at the non-salient regions. This can be seen evidently from the binary threshold selection. Compared with the threshold 0.7 in Rahtu's method, our threshold is only 0.1. This means the saliency values at the non-salient regions have been inhibited strongly.

6   Conclusion

In this paper, we firstly define two types of salient stimuli followed Weber's Law. One type is local stimulus, and the other type is global stimulus. Then a center-surround computational model with ring topology structure is designed to extract salient responses. Finally, we discuss a soft-decision integration strategy of local and global saliency in CIELAB color space. Experiments demonstrate our method has stronger response in activation regions and better inhibition performance in other regions.

References

 1. Itti, L., Koch, C., Niebur, E.: A Model of Saliency-Based Visual Attention for Rapid Scene Analysis. IEEE Trans. PAMI 20, 1254–1259 (1998)
 2. Gao, D., Vasconcelos, N.: Discriminant Saliency for Visual Recognition from Cluttered Scenes. In: Neural Information Processing Systems, pp. 481–488 (2004)
 3. Navalpakkam, V., Itti, L.: An Integrated Model of Top-down and Bottom-up Attention for Optimizing Detection Speed. In: IEEE Conf. Computer Vision and Pattern Recognition, pp. 2049–2056 (2006)
 4. Achanta, R., Estrada, F., et al.: Salient Region Detection and Segmentation. In: Gasteratos, A., Vincze, M., Tsotsos, J.K. (eds.) ICVS 2008. LNCS, vol. 5008, pp. 66–75. Springer, Heidelberg (2008)
 5. Achanta, R., Hemami, S., et al.: Frequency-tuned Salient Region Detection. In: IEEE Conf. Computer Vision and Pattern Recognition, pp. 1597–1604 (2009)
 6. Gopalakrishnan, V., Hu, Y., Rajan, D.: Salient Region Detection by Modeling Distributions of Color and Orientation. IEEE Trans. Multimedia 11(5), 892–905 (2009)
 7. Zhang, Q., Liu, H., et al.: An Improved Computational Approach for Salient Region Detection. Journal of Computers 5(7), 1011–1018 (2010)
 8. Gao, D., Mahadevan, V., Vasconcelos, N.: Discriminant Saliency, the Detection of Suspicious Coincidences, and Applications to Visual Recognition. IEEE Trans. PAMI 31(6), 989–1005 (2009)
 9. Packer, O.S., Dacey, D.M.: Synergistic Center-surround Receptive Field Model of Monkey H1 Horizontal Cells. Journal of Vision 5(11), 1038–1054 (2005)
10. Deco, G., Heinke, D.: Attention and Spatial Resolution:Theoretical and Experimental Study of Visual Search in Hierarchical Patterns. Perception 36(3), 335–354 (2007)

11. Vazquez, E., Gevers, T., et al.: Saliency of Color Image Derivatives: A Comparison Between Computational Models and Human Perception. Journal of the Optical Society of America A 27(3), 613–621 (2010)
12. Liu, T., Sun, J., et al.: Learning to Detect A Salient Object. IEEE Trans. PAMI 33(2), 353–367 (2011)
13. Rahtu, E., Heikkil, J.: A Simple and Efficient Saliency Detector for Background Subtraction. In: IEEE Intl. Workshop on Visual Surveillance, pp. 1137–1144 (2009)

Multi-scale Image Analysis
Based on Non-Classical Receptive Field Mechanism

Hui Wei, Qingsong Zuo, and Bo Lang

School of Computer Science,
Fudan University,China
{weihui,09210240055,09110240025}@fudan.edu.cn

Abstract. In the real world, the biological visual system is more efficient than the machine visual system in analyzing visual information. Physiology theories show that this efficiency owes to the multi-layer neural network in human visual system, in which every layer accomplishes different tasks and is related with other layers. The low-level stages of the human visual system, especially the retina, can provide certain scale information for the high-level stages of visual system through using the non-classical receptive field (nCRF) mechanism. This mechanism that the nCRF size can be adjusted automatically by ganglion cell (GC) can achieve a multi-scale image analysis. The results, reflecting the distribution of the image information, can be shared by several algorithms or processes solving different visual tasks, such as contour detection and image segmentation. A model of multi-scale image analysis based on GC has been proposed in this paper, which retains the key information and reduces the redundancy information for the further stages of the visual system. Experimental results on N-cut and contour detection show that this multi-scale image analysis model provides distinctive improvement for these image processing tasks.

Keywords: multi-scale, image analysis, contour detection, nCRF mechanism.

1 Introduction

In recent decades, multi-scale image analysis has become one of the most effective approaches to extract useful information from images and is widely applied in image denoising, contour detection, image segmentation, object recognition and many other tasks [1-6]. However, most of the multi-scale transforms such as cosine transform and wavelet transform cannot meet all of the five requirements – multi-resolution, locality, orientation, anisotropy and critical sampling, which should be met by an optimal image analysis, according to the physiology studies [7]. Fortunately, the human visual system is able to satisfy all the requirements, which gives us a great inspiration for designing a physiologically plausible multi-scale analysis model.

In the human visual system, a pyramid hierarchical structure, information is mainly processed from the retina through the lateral geniculate nucleus to the visual cortex. As the first stage of information perception and transformation in system, the retina

obtains illumination, color and spatial information and transfers them to higher processing phases. Ganglion cells are the final output neurons of the vertebrate retina. Since 1960s, many researchers [8, 9] have found that a large-scope area beyond the classical receptive field (CRF) of a ganglion cell, named non-classical receptive field (nCRF), can exert modulatory effects on the response of the stimuli in the CRF. Light spot stimuli in nCRF cannot cause a direct reaction of the cell, but they can facilitate, inhibit or disinhibit the behavior of that cell. Li Chaoyi et al. studied the nCRF spatial summation property of disinhibitory of cat retinal ganglion cells [10]. They found that disinhibitory nCRF can not only enhance the border contrast but also compensate the loss of low-frequency which was caused by the antagonistic center/surround mechanism of the CRF. So, nCRF mechanism plays an important role in transmitting the image information of luminance contrast and luminance gradient in local area [11]. Likewise, there are also nCRFs of majority neurons in the primary visual cortex (V1), which allows neurons to integrate information to participate in many complex perceptual tasks such as contour integration and surface segmentation [12-14].

Based on the insight of nCRF multi-scale analysis and inhibition mechanism, several biology-based image processing models have been presented recently [15-18]. Although, these models have good performance in some aspects compared to traditional image processing approaches, there are still some shortages in practice. These models only involve nCRF mechanism in V1, ignoring the function of retina in the procedure of information processing. For example, Grigorescu et al. in [15] proposed a model of contour detection in which many parameters are to be set experimentally. In the model, a parameter-- standard deviation of the Gaussian factor, is closely relative to every given image and is difficult to be pre-set. In recent, many researchers [19-21]have proposed some multi-scale contour detection models as well that can improve results to certain extent, but the pre-estimation of scale needs prior knowledge about the image and is a difficult task.

To overcome the problem of parameters selection, this paper proposes an nCRF multi-scale analysis model involving cortical cells and retinal cells that transmit scale information to higher layers of the visual system. This model is able to solve the problems mentioned above. To prove that our model can facilitate image analysis, two experiments are designed. (1) N-cut[22] segmentations on images that are preprocessed by our model are comparable to N-cut segmentations on original images. Experimental results show that the multi-scale image analysis model can exhibit stable improvement for image segmentation based on N-cut. (2) Contour detection approach involving our model of nCRF of ganglion cells, simple cells and complex cells is compared to the model in [15].

This paper is organized as follows. Section 2 presents our multi-scale image analysis model based on multi-level nCRF, which describes the computation model of biological visual system for excitability and inhibition actions in transmission process from low visual system--retina to high visual system. Experimental results on the contour detection and N-cut from above scheme are presented in Section 3. In section 4, a summary of the results and a conclusion are given.

2   Algorithm

Fig. 1 depicts the structure of retinal ganglion cell RF (including nCRF), in which the RF of ganglion cell embodies CRF (CRF center and CRF Surround) and nCRF. The width of ring-formed nCRF is larger than the radius of CRF, as shown in Fig. 1. In section 2.1, the response of GC is calculated using a 3-Gaussian model. Section 2.2 describes the algorithm of dynamically adjusting the size of RF and RFs of different size in line with different scales of wavelet transform as well. Taking the response and size of RF as input, a contour detection model based on multi-layer nCRF in human visual system is addressed in section 2.3.
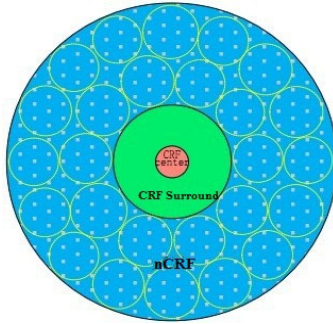
Fig. 1. The structure diagram of GC's RF. The RF contains three parts—the CRF center, the inner circle area, the CRF surround, the intermediate annulus, and the nCRF, the extern ring area. Small circles on nCRF are subunits of the nCRF.

2.1   Response of GC

According to the neuroscience, the response of GC to stimuli within RF can be well described by a 3-Gaussian function[23].The detailed procedure of computation can be expressed as follow:

$$GC(x, y) = (I(x, y) * W_{CRF})(x, y) + \log_2 \left(1 + I(x, y) * W_{nCRF}(x, y)\right) \quad (1)$$

where $GC(x, y)$ is the response of a ganglion cell; $I(x, y)$ is the image stimulus within RF; W_{CRF} and W_{nCRF} are the weighting functions of the CRF, and the disinhibitory nCRF, respectively. They are defined as:

$$W_{CRF} = \frac{A_1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(x-x_0)^2+(y-y_0)^2}{2\sigma_1^2}} - \frac{A_2}{\sqrt{2\pi}\sigma_2} e^{-\frac{(x-x_0)^2+(y-y_0)^2}{2\sigma_2^2}} \quad (2)$$

$$W_{nCRF} = \frac{A_3}{\sqrt{2\pi}\sigma_3} e^{-\frac{(x-x_0)^2+(y-y_0)^2}{2\sigma_3^2}} \quad (3)$$

A_1, A_2 and A_3 are the sensitivity of the CRF center, the CRF surround and the nCRF respectively. σ_1, σ_2 and σ_3 are the radii of the CRF center, the CRF surround and the nCRF respectively and can be adjusted adaptively according to the stimulus; x_0 and y_0 are the center coordinates of the RF. In actual computation, $\sigma_3 = 4\sigma_2$, $\sigma_2 = 5\sigma_1$, $A_1 = 1$, $A_2 = 0.18$, $A_3 = 0.05$ according to the physiology theory [24].

2.2   Size of RF

A mathematical model is shown in Fig. 2. The philosophy of this model is as follows: when a RF receives a uniform stimulus, the RF expands continuously; otherwise, the RF shrinks drastically. As the nCRF consists of many sub-regions, each of which has its own stimulus unit. The variance of stimulation in each sub-region is calculated. It is used to judge whether the stimulus is homogeneous and then decide to make the RF expand or shrink. The size of RF, radius of RF, is determined and transmitted to a later processing stage.
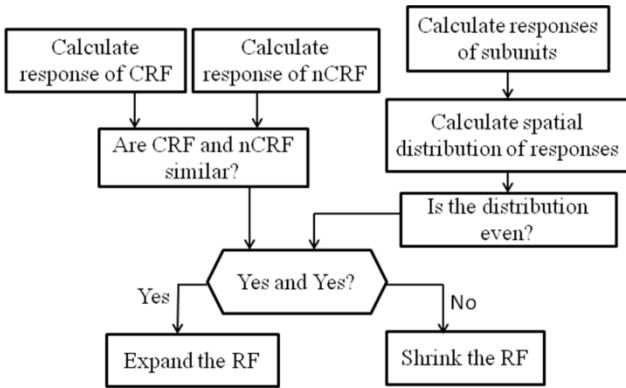
Fig. 2. Self-adaptive model of RF

2.3   Multi-scale Contour Detection Algorithm

In computer vision and image processing, detection of edges and contours are important tasks in object recognition and other applications. Here, we proposed a model involving retinal GCs, cortical simple cells and complex cells. The model provides multi-scale analysis and achieves contour detection efficiently on a wide range of images adaptively. This model can address efficiently the problem of choosing an appropriate scale in [15]. Fig. 3 shows the schematic drawings illustrating the general flowchart of the proposed model for multi-scale contour detection. Unlike previous techniques based on nCRF, this system does not need to specify the value of the standard deviation σ --scale parameter and know any other prior information.
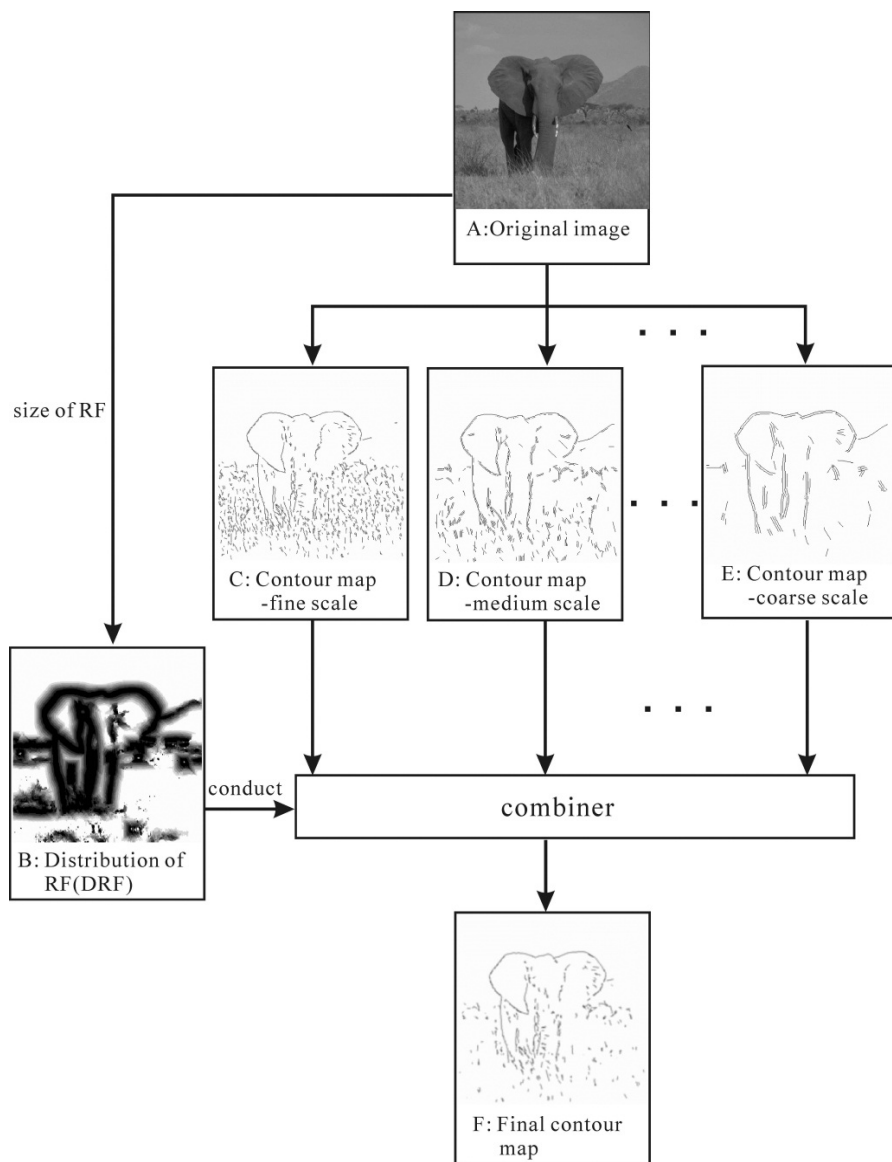
Fig. 3. The proposed model--multiscale contour detector based on nCRF. (A) Original image. (B) DRF which is computed by the algorithm—self-adaptive RF in section 2.2. (C) Contour map at fine scale is computed by the algorithm in [15] with the parameter σ which is equal to the minimum value of size of RF in DRF. (D) Contour map at medium scale with the medium scale σ. (E) Contour map at coarse scale with the maximum scale σ. (F) Final contour map which is combined by the series of contour maps at different scales.

In short, given an original image (Fig. 3A), distribution of RF can be calculated from the algorithm described in section 2.2, the darker pixels in Fig. 3B representing the finer scale, similarly, the brighter the coarser. Contour detectors [15] with different scales are used to extract a series of contour maps (Fig. 3C, D, E). Some object contours or parts of them which are missing for some coarse-scale detectors are well detected for other fine-scale detectors. Likewise, many tiny edge details that are not object contours and are obtained by fine-scale detectors can be ignored by coarse-scale detectors. This observation suggests that the combination of the series of contour maps at different scales will retain all object contours and suppress minor edge details. A series of contour maps can be extracted by the algorithm [15], and the next step is to combine the contour maps. Carefully observing the Fig. 3B, we can find that the regions in which object contours exist are darker than the regions in which tiny edges exist. Thus, according to the distribution of size of RF in DRF, we can construct the combiner that selects parts from the fine-scale contour maps in dark regions of DRF and chooses parts from the coarse-fine contour maps, producing the final contour map.

In previous works [15, 18, 23, 24], the models detected object contours through selecting the best result from many contour maps at different scale. Comparing to our model, other models cost lots of manual labor and require selecting an appropriate scale that is difficult to pre-estimate. In our approach we apply an N-level multi-scale analysis in order to remove the littery texture still present in contour map at fine scale and retain salient object contours in contour map at coarse scale. Object contours are present continuously at fine scales, while texture disappears at coarse scales. So, we can combine a series of contour maps at different scales. The final contour map is composed by the corresponding parts of the series of contour maps at different scales according to the brightness in DRF. For example, parts of the contour map at fine scale that is dark in the same regions of DRF, are retained to compose the according parts of the final contour map. However, parts of contour map at coarse scale that is bright in DRF are reserved to compose the final contour map.

3   Experimental Results

This section presents some experimental results. GCs construct the distribution of different-size RFs according to the input image. The response and size of GCs are used for image segmentation and contour detection.

3.1   Distribution of RFs and Segmentation on GC

Fig. 4 displays the distribution of the RFs of different size. From Fig. 4, we can observe that small RFs mainly cover the contours, and large-size RFs gather together on the texture and roughly unchanged area. This result shows that RFs in GC layer can obtain the scale information and variation of color or intensity.

Through the 3-Gaussian model operating, the RFs in GC layer can also highlight the object while suppressing the background, which can be proved by the N-cut results, as Fig. 5.
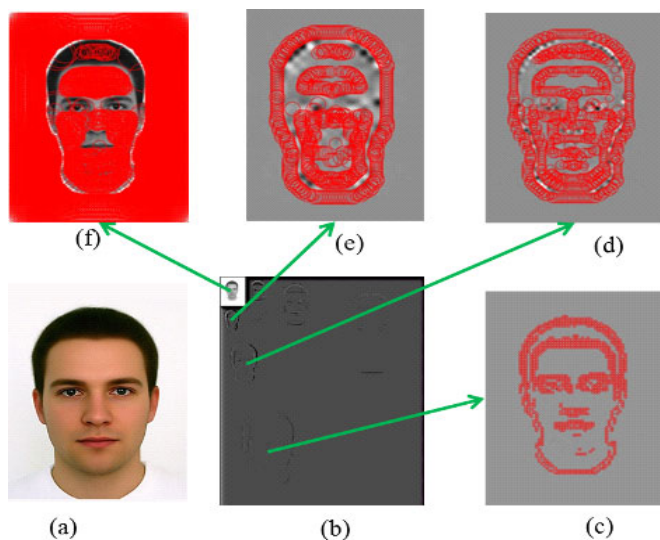
Fig. 4. Distributions of RFs with different scales consist with the Wavelet Transform. Red circles denote RFs with different scales. (a) Original image. (b) Result of wavelet transform. (c) Distribution of minimum-size RFs, which represents details of the face. (d) Distribution of medium-size RFs, which closely matches the middle level wavelet transformation. (e) Distribution of large-size RFs, which closely matches the third level wavelet transformation. (f) Distribution of full-size RFs, which represents the background of the image.
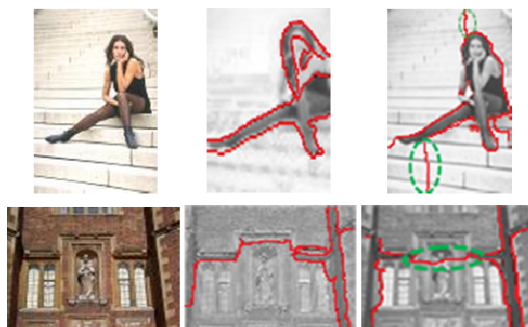
Fig. 5. Results of testing N-cut on outputs of GC and original images. Left column is the original images; middle column is N-cut on output of GC and right column is N-cut on original image. Green ellipses mark the unsatisfied segmentations.

The N-cut [22] results are shown in the middle column, executing N-cut algorithm on the response of GC. The original images are shown in the left column. The results of running N-cut on the original image are shown in the right column. The green ellipses mark the specific places in which the segmentations are improved by our model. This two comparative experimental results show that the nCRF model can facilitate image segmentation, specially, in separating object from background.

3.2   Contour Detection

In the experiment of contour detection, we compare the results of our model with the model in [15]. The first and second rows list the original input images and corresponding ground truth images in Fig. 6, respectively. The third and fourth rows show the best results of the contour detectors with the anisotropic inhibition [15] and our contour model, respectively. These images used in the experiment are acquired from the web site: http://www.cs.rug.nl/~imaging/databases/contour_database/.

(a) Hyena   (b) Elephant   (c) Goat   (d) Rino
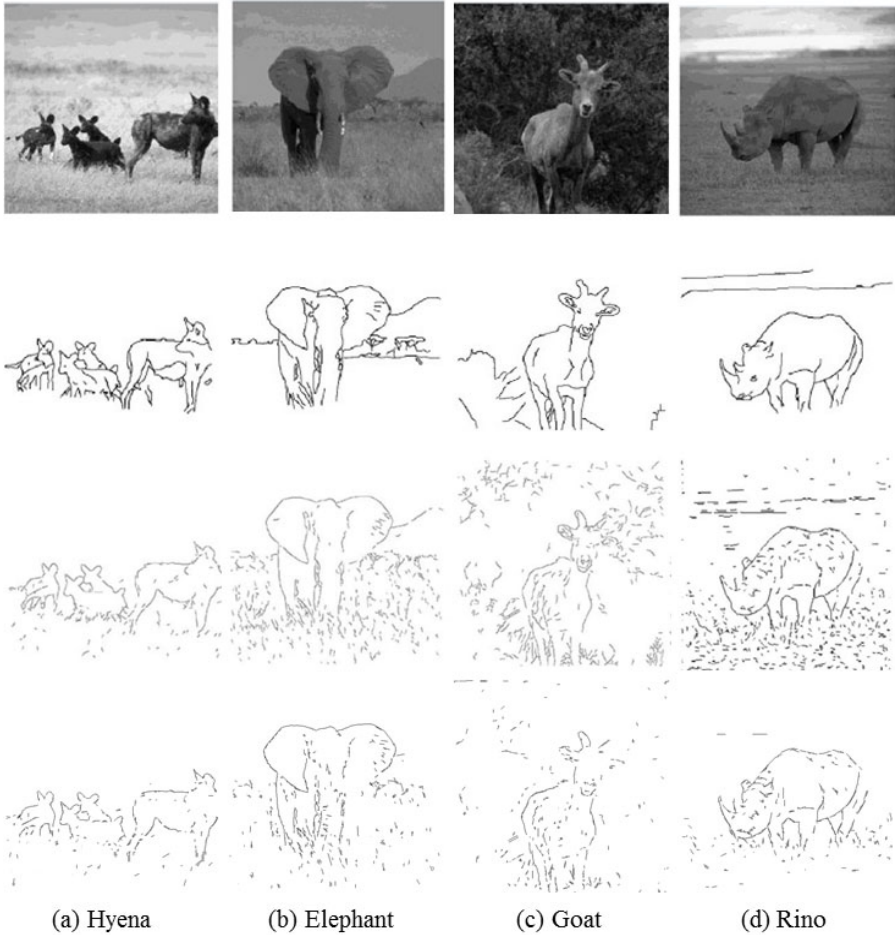
Fig. 6. Comparison of contour detection results for different schemes. From top to down, the above images are the original images, the corresponding ground-truth contours, the best contour maps in [15] and our results, respectively.

The proposed multi-scale contour detector contributes to a better contour detection mainly by applying different scales inhibition to different contexts such as fine scale on object contours and coarse scale on texture edges. The results clearly show that our

computational model suppresses the edges originating from textural regions much more effectively than the inhibition-based model [15]. Our results are of fewer trivial edge fragments in the textural regions of the images shown in Fig. 6. In addition, our model produces a result that is quite close to the ground truth for the rhinoceros image shown in the first column of Fig. 6.

Note: In the third row, the goat, elephant and hyena contours are from the paper of Grigorescu et al. [15] and the rino contour is from their given website: http://matlabserver.cs.rug.nl/edgedetectionweb/web/index.html.

4   Conclusions

The multi-scale image analysis model presented in this paper can obtain scale information and illumination intensity from fine scale to coarse scale using the nCRF mechanism. We show that an adaptive algorithm that attempts to find different scale information for natural scenes will develop a family of localized, bandpass non-classical receptive fields, similar to those found in the ganglion cells. According to the results, it can be concluded that in some high-level image process tasks, such as contour detection and image segmentation, this model plays a very useful role in their performance improvement.

References

1. Portilla, J., Strela, V., Wainwright, M.J., Simoncelli, E.P.: Image denoising using scale mixtures of Gaussians in the wavelet domain. IEEE Transactions on Image Processing 12, 1338–1351 (2003)
2. Guanghong, H., Yingjun, P., Wei, J.: An Improved Image Denoising Method Based on Multi-scale Correlation in Wavelet Domain. In: Conference An Improved Image Denoising Method Based on Multi-scale Correlation in Wavelet Domain (2006)
3. Mahmoodi, S., Sharif, B.S., Chester, E.G.: Contour detection using multi-scale active shape models. In: Conference Contour Detection using Multi-scale Active Shape Models, vol. 702, pp. 708–711 (1997)
4. Tremblais, B., Augereau, B.: A fast multi-scale edge detection algorithm. Pattern Recognition Letters 25, 603–618 (2004)
5. Yang, W., Qianqian, W.: Image segmentation based on multi-scale local feature. In: Conference Image Segmentation Based on Multi-scale Local Feature, pp. 1406–1409 (2010)
6. Wei, Z., Hongli, D., Dietterich, T.G., Mortensen, E.N.: A Hierarchical Object Recognition System Based on Multi-scale Principal Curvature Regions. In: Conference A Hierarchical Object Recognition System Based on Multi-scale Principal Curvature Regions, pp. 778–782 (2006)

7. Donoho, D.L., Flesia, A.G.: Can Recent Innovations in Harmonic Analysis Explain Key Findings in Natural Image Statistics? Network: Computation in Neural Systems 12, 371–393 (2001)
8. Ikeda, H., Wright, M.: The Outer Disinhibitory Surround of the Retinal Ganglion Cell Receptive Field. The Journal of Physiology 226, 511 (1972)
9. Krüger, J., Fischer, B.: Strong Periphery Effect in Cat Retinal Ganglion Cells. Excitatory Responses in ON- and OFF-center Neurones to Single Grid Displacements. Experimental Brain Research 18, 316–318 (1973)
10. Chao-Yi, L., Wu, L.: Extensive Integration Field Beyond the Classical Receptive Field of Cat's Striate Cortical Neurons–Classification and Tuning Properties. Vision Research 34, 2337–2355 (1994)
11. Chao-Yi, L., Xing, P., Yi-Xiong, Z., Hans-Christoph, M.: Role of the Extensive Area Outside the X-Cell Receptive Field in Brightness Information Transmission. Vision Research 31, 1529–1540 (1991)
12. Jones, H., Grieve, K., Wang, W., Sillito, A.: Surround Suppression in Primate V1. Journal of Neurophysiology 86, 2011 (2001)
13. Kapadia, M.K., Westheimer, G., Gilbert, C.D.: Spatial Distribution of Contextual Interactions in Primary Visual Cortex and in Visual Perception. Journal of Neurophysiology 84, 2048 (2000)
14. Walker, G.A., Ohzawa, I., Freeman, R.D.: Suppression Outside the Classical Cortical Receptive Field. Visual Neuroscience 17, 369–379 (2000)
15. Grigorescu, C., Petkov, N., Westenberg, M.A.: Contour Detection Based on Nonclassical Receptive Field Inhibition. IEEE Trans. Image Process. 12, 729–739 (2003)
16. Papari, G., Campisi, P., Petkov, N., Neri, A.: IEEE: Contour Detection by Multiresolution Surround Inhibition. In: Proceedings International Conference on Image Processing, vol. 1-7, pp. 749–752 (2006)
17. Fernandes, B.J.T., Cavalcanti, G.D.C., Ren, T.I.: Nonclassical Receptive Field Inhibition Applied to Image Segmentation. Neural Network World 19, 21–36 (2009)
18. Zeng, C., Li, Y., Yang, K., Li, C.: Contour Detection Based on a Non-Classical Receptive Field Model with Butterfly-Shaped Inhibition Subregions. Neurocomputing 74, 1527–1534 (2011)
19. Sivaswamy, G.D.J.J.: Multi-scale Approach to Salient Contour Extraction. In: Conference Multi-scale Approach to Salient Contour Extraction, pp. 186–193 (2005)
20. Papari, G., Petkov, N.: An Improved Model for Surround Suppression by Steerable Filters and Multilevel Inhibition with Application to Contour Detection. Pattern Recognit. 44, 1999–2007 (2011)
21. Papari, G., Petkov, N.: Edge and Line Oriented Contour Detection: State of the Art. Image and Vision Computing 29, 79–103 (2011)
22. Shi, J., Malik, J.: Normalized Cuts and Image Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 888–905 (2000)
23. Ghosh, K., Sarkar, S., Bhaumik, K.: Image Enhancement by High-Order Gaussian Derivative Filters Simulating Non-classical Receptive Fields in the Human Visual System. In: Pal, S.K., Bandyopadhyay, S., Biswas, S. (eds.) PReMI 2005. LNCS, vol. 3776, pp. 453–458. Springer, Heidelberg (2005)
24. Chaoyi, Q.F.L.: Mathematical Simulation Of Disinhibitory Properties Of Concentric Receptive Field. Acta Biophysica Sinica 11, 214–220 (1995)

Visual Constructed Representations for Object Recognition and Detection

Yasuomi D. Sato[1,2,3] and Yasutaka Kuriya[1]

[1] Graduate School of Life Science and Systems Engineering,
Kyushu Institute of Technology,
2-4 Hibikino, Wakamatsu-ku, Kitakyushu 808-0196, Japan
sato-y@brain.kyutech.ac.jp
[2] Frankfurt Institute for Advanced Studies (FIAS),
Johann Wolfgang Goethe University,
Ruth-Moufang-Str. 1, Frankfurt am Main, 60438, Germany
sato@fias.uni-frankfurt.de
[3] BSI-TOYOTA Collaboration Center, Brain Science Institute, RIKEN
2-1 Hirosawa, Wako, Saitama, 351-0198, Japan

Abstract. We propose a neurally inspired model for parallel visual process for recognition and detection. This model is based on the Gabor feature explicit representation construction. An input image is decomposed of different scale features through the low-pass filter. Nevertheless, recycling and overlapping again the scale features, the most likely object stored in memory can be detected on the input image. This is done by scale feature correspondence finding. Simultaneously, Gabor feature representations stored in memory are also constructed by selecting the most similar scale features to the input. We also test a recognition ability of our model, using a number of facial images of different persons. Distortion invariant recognition is also demonstrated.

Keywords: Visual Object/Face Recognition and Detection, Gabor Feature Representation Construction.

1 Introduction

In research fields of image recognition and pattern recognition, there were many developmental systems for object recognition and object detection, which have so far been proposed. The representative ones are the scale invariant feature transformation (SIFT) feature detection [1] and the adaboost for face detection developed by Viola and Jones [2], which have high potentials toward real applications onto various technological aspects, because they possess many of high vision processing techniques. However, they are loosely in neural style.

Progress of neuroscientific knowledge about the visual information processing is expected as essential for more advanced technology innovation for future computer vision. The vision process in the brain is very complicated and sophisticated. It is extraordinarily difficult to solve and understand the vision mechanism

only with cultivating wisdoms until now. We will have to extract a not-so-simple but central mechanism from complicated vision processing in the brain, in order to redevelop the core vision process on computers, or on hardware.

One of neurophysiologically plausible and advanced recognition systems is the HMAX model proposed by Riesenhuber and Poggio [3]. The basic HMAX model consists of a hierarchy of five levels, from the S1 layer with simple-cell like response properties to the higher layer with view-tuned cell like response properties for shape tuning and invariance. It is a very similar computational model of object recognition in cortex to the Neocognitron [4]. Though the HMAX model would rather be physiologically plausible, but it has also a heavily computational cost problem. Thus, the HMAX model might still be unsuitable for practical application to industrials.

In recent years, propaganda for "Vision Reconstruction" is being spread. The propaganda is based on a physiological percept representation concept [5]. This conceptual idea is even taken into account the practical application in the near future.

But the original concept of the Vision Reconstruction follows the Dynamic Link Architecture (DLA) [6] with the help of a concrete application, a face recognition system that has been highly successful in industrial applications. The main mechanism in the DLA is to achieve object recognition by constructing segmentations of the representation stored in memory. This was established with a correspondence-based recurrent network model [7]. In this network model, a so-called macrocolumnar model for representing cortical activity dynamics is employed, supporting functionalities of recognition and detection for object. However the network model is very complicated and gets easily into difficulties to understand the mechanism. There are still ample rooms to discuss about developmental improvement toward the practical application.

In this work, with full use of neuroscientific knowledge, we propose an improved version of elastic graph matching for simultaneous object recognition and detection, extracting the essences from the aforementioned recurrent network model. Then, we will test the system ability and discuss improvements or advantages of the system proposed here, toward the implementation into digital hardware circuits.

2   System Design Concept

In order to model a concept of "Gabor Feature Construction" proposed in this work, we shall begin with by explaining a system for simultaneous detection and recognition of a single object (or, multi objects), which is in a neural style.

2.1   Input Signal

Firstly, let us assume that visual information on the retina incoming from eyes are established by a number of overlapping filters that are selectively set up with different ranges of spatial frequency and orientation. This assumption is based on both results of physiological experiments and image processing theory.
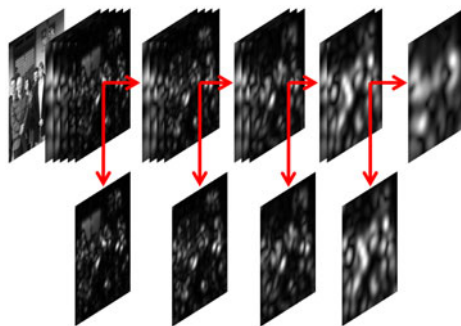
Fig. 1. Multi-scale Gabor filter decomposition. The high spatial frequency elements of visual input information are sequentially removed with a low-pass Gabor filter in the bottom-up flow.

In physiological experiments for receptive field, it is already found that neurons being tuned to high spatial frequencies have narrower tuning range than neurons being tuned to low spatial frequencies. Also, the receptive field structure, which can often be called a Gabor function, was observed as constructed by multiplying a global sinusoidal grating by a bell-shaped Gaussian envelope [8].

Such overlapping filters are, of course, widely acceptable as a series of multiscale transforms in applications to signal and image processing. In signal analysis by using a multi-scale transform of *Gabor function*, the signal (namely, visual information) presents a set of features and structures occurring at different spatial frequencies of Gabor function [9].

Furthermore, vision of the human eyes can be conceived to be achieved through low pass filter processing [10]. Once an input image about some environmental scene is received on a retina, the highest spatial frequency component of the Gabor filter is sequentially discarded in bottom-up flow as shown in Fig. 1. The prospective discarding spatial frequency elements may be stored in another area through another pass in the visual cortex. The inferior temporal (IT) cortex can be considered as one of candidates.

2.2 Memory Representations and Recognition

Next, we consider memory representation of visual image projected on a retina. Just as mentioned before, probably, an input image may be dispersed in fine pieces of the scale and orientation, through dynamically continuous low-pass filters. Each of the pieces will then be stored as memory in different higher areas of a visual cortex.

Modeling of such memory storage of Gabor feature representations is significantly seen in developmental face recognition systems using Elastic Bunch Graph Matching. The main concept is given by the following: We may easily memorize apparently unique parts of a face such as a tip of a nose, eyes, ears, etc. The different facial parts of each person, which can sometimes be called the
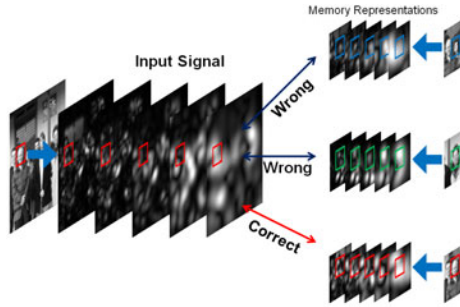
Fig. 2. A correspondence based recognition concept. Input signals (I) or representations stored in memory (M) are broken up in fractions, in terms of a spatial frequency level of Gabor feature. For each spatial frequency level, correct or wrong correspondences are found between the I and M. Here let us assume that graphs are set up within color squares.

fiducial points, may separately be stored as feature representations in memory. Furthermore, the feature representation of each fiducial point represents a set of orientation and spatial frequency components of the Gabor filter.

The stored memory representations are completely disconnected, in terms of Gabor filters, facial parts and their personality. But, such disconnected Gabor feature representations for each fiducial point will be integrated again once the fiducial point on a face, perceived on a retina, looks for and finds the most likely Gabor feature in memory. The input face can therefore be recognized as it is once complete reunion of fiducial points of the input face is achieved in memory representations.

In description of stored memory representations, not only aforementioned face graphs, but also square grid ones (such as Elastic Graph Matching (EGM)) should be employed. In any case using square grid graphs, recognition would be succeeded when each grid on the memory graph can detect the most similar feature on the corresponding grid on the input graph (Fig. 2).

This correspondence finding is one of the philosophies of the dynamic link architecture, which can sometimes be called as "graph topology preservation" [6]. The graph topology preservation means that all potential correspondences are established by correct links between two graphs when a set of links are consistently preserved with the topology, giving high similarities.

Unfortunately, in realistic application, such graph topology preservation can difficultly be seen. Nevertheless, a set of links takes incorrectly the high similarities. It is wrong correspondence. We may have to be always care of such correspondence problem [7].

2.3 Gabor Filter Construction: Additional Object Detection

We will have to know an additional mechanism how to detect a retinal position of an input object recognized as one of objects stored in memory. Let us assume
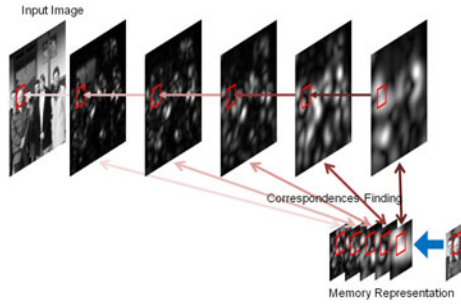
Fig. 3. A correspondence based object detection concept. Correct scale feature correspondence is found between Input and Memory, in order to detect the most suitable region for memory representation on the RF of each spatial frequency level. This is done in the top-down as well as in the coarse-to-fine.

that as mentioned in Sects. 2.1 and 2.2, input signals and memory representations are respectively decomposed of the different spatial frequency levels.

As the first step, the lowest spatial frequency level Gabor feature for memory representations is wandering around receptive fields (RFs) for the same lowest level of input signals, to search the position that has the most similar feature to the one for memory representations. In order to find such a position, a neural inspired version of the so-called maximum operation is employed, which is under an assumption that neurons in a layer compete with each others for activation. We will thoroughly explain the maximum operation method in the next section.

Once the most suitable position for similarity to memory feature representation is detected, it is projected to the corresponding position on a RF for the next lower level. Analogous to the lowest level case, the most appropriate position must be found by the similarities to memory feature with the next lowest levels. Such top-down detection process must be repeated until being projected to the retina (see, Fig.3).

We should not here forget reusing input signals for the sake of integration of the feature pieces into a whole explicit representation for the Gabor feature. It is also analogous for the corresponding memory representation. What we address is Gabor feature construction for both the memory representation and input signal, achieving simultaneous visual object recognition and detection.

3   A Network Model and Simulations

We propose a neural style of Gabor feature construction model for parallel processing of object recognition and detection, which is grounded on the neural concept mentioned in Sect. 2. If we straightforwardly model the neural concept in Sect.2, one can easily expect that there is heavily computational cost for Gabor filtering, because as the Gabor kernel size is bigger, in addition, an image size is also larger, naturally, it takes a longer and longer time to filtering an image. In order to escape from such a heavily computation cost problem, a Gabor

pyramid method is employed. The essence of the Gabor pyramid utility is the scale Gabor feature correspondence.

3.1 Gabor Wavelet

A Gabor feature can be extracted at a pixel position $\boldsymbol{p} = (x, y)$ on a natural grayscale image through convolutions of the image F with a family of Gabor functions $\psi_r(\boldsymbol{p})$. r $(= 0, \ldots, 7)$ is an orientation parameter. The Gabor feature consists of 8 different orientation components. Each orientation component is given by a convoluted value $(\hat{J}_r)$:

$$\hat{J}_r(\boldsymbol{p}_l) = \int F(\boldsymbol{p}_l - \hat{\boldsymbol{p}}_l) \psi_r(\boldsymbol{p}_l - \hat{\boldsymbol{p}}_l) d^2 \hat{\boldsymbol{p}}_l, \tag{1}$$

$$\psi_r(\boldsymbol{p}) = \frac{|\boldsymbol{k}_r|^2}{\sigma^2} \exp\left(-\frac{|\boldsymbol{k}_r|^2 |\boldsymbol{p}|^2}{2\sigma^2}\right) \left[\exp\left(i\boldsymbol{k}_r \cdot \boldsymbol{p}\right) - \exp\left(-\frac{\sigma^2}{2}\right)\right]. \tag{2}$$

r is explicitly given as described below in (3). $\sigma = 2\pi$. The wave number vector $\boldsymbol{k}_r$ can be expressed as

$$\boldsymbol{k}_r = \begin{pmatrix} k_{r,x} \\ k_{r,y} \end{pmatrix} = \begin{pmatrix} k \cos\phi_r \\ k \sin\phi_r \end{pmatrix}, \qquad k = \frac{k_{\max}}{k_0}, \qquad \phi_r = \frac{\pi}{8} r, \tag{3}$$

where $k_{\max} = \pi/2$ and $k_0 = \sqrt{2}$. One orientation component J_r in the Gabor feature takes a real part of $\hat{J}_r$:

$$J_r(\boldsymbol{p}) = \mathrm{Re}\left(\hat{J}_r\right). \tag{4}$$

3.2 Scale Feature Correspondence Findings

Let us consider $k_{\max}/k_0^l$ where l $(= 0, 1, 2, 3, 4)$ is a spatial frequency parameter. This enables us to control the kernel size of the Gabor function. The kernel size is bigger when the spatial frequency parameter is larger. We prepare one image with a certain original size I_0. Then, the original image is continuously down-sampled with $[1/k_0]^{2l}$. Let each down-sampled image be I_l.

For the spatial frequency level l of the Gabor filter, we obtain the Gabor feature set $\boldsymbol{J}^{I_0}(k_0^l)$ when the image I_0 of the original size is used. Meanwhile we also get the Gabor feature set $\boldsymbol{J}^{I_l}(0)$ for the level $l = 0$ with the rescaled image I_l. They have scale correspondence with each other as follows:

$$\boldsymbol{J}^{I_0}(k_0^l) \approx \boldsymbol{J}^{I_l}(0). \tag{5}$$

The scale feature correspondence is already founded as the multi-scale in scale-space approaches [11]. As shown in (5), an original size image, Gabor-filtered with an individual spatial frequency level, is approximated to the correspondingly sub-sampled image smoothed with the low-pass filter. This is well-known as used, not only for effectively reducing computational costs in the Gaussian filtering or its Laplacian, but also for stored memory feature size [12].

3.3 Elastic Graph Matching

A number of natural grayscale images of different object with an arbitrary pixel are prepared in memory (M_0^g). The size of each image is rescaled with $(1/k_0)^{2l}$ (called M_l^g). A square graph of $(2N+1) \times (2N+1)$ nodes is set on each rescaled image (Here $N = 4$). It is notices that pixel position for each node is also rescaled with $(1/k_0)^{2l}$.

Let us define an energy function E_l between two graphs of the input and model images:

$$E_l(\hat{\boldsymbol{p}}_l) = \sum_{p_l \in \aleph_l} e_{p_l}^s(\hat{\boldsymbol{p}}_l) - \lambda_d e_{p_l}^d(\hat{\boldsymbol{p}}_l), \qquad (6)$$

$$e_{p_l}^s(\hat{\boldsymbol{p}}_l) = \frac{\sum_r J_r^{I_l}(\hat{\boldsymbol{p}}_l) \cdot J_r^{M_l}(\boldsymbol{p}_l)}{\sqrt{\sum_r \left(J_r^{I_l}(\hat{\boldsymbol{p}}_l)\right)^2 \sum_r \left(J_r^{M_l}(\boldsymbol{p}_l)\right)^2}}, \qquad (7)$$

$$e_{p_l}^d(\hat{\boldsymbol{p}}_l) = \frac{1}{|\aleph_l|} \sum_{p_l \in \aleph_l} \frac{\left|\sum_{p_l' \in \aleph_{p_l}} D_{p_l,p_l'}^{M_l} - \sum_{p_l' \in \aleph_{p_l}} D_{\hat{p}_l,p_l'}^{I_l}\right|}{\left|\sum_{p_l' \in \aleph_{p_l}} D_{p_l,p_l'}^{M_l}\right|}$$

$$+ 1 - \frac{1}{|\aleph_l|} \sum_{p_l \in \aleph_l} \frac{\left|\sum_{p_l' \in \aleph_{p_l}} A_{p_l,p_l'}^{M_l} \cdot \sum_{p_l' \in \aleph_{p_l}} A_{\hat{p}_l,p_l'}^{I_l}\right|}{\left|\sum_{p_l' \in \aleph_{p_l}} A_{p_l,p_l'}^{M_l}\right| \cdot \left|\sum_{p_l' \in \aleph_{p_l}} A_{\hat{p}_l,p_l'}^{I_l}\right|} \qquad (8)$$

Here $\aleph_l$ is a set of all nodes on the graph. $e_{p_l}^s(\hat{\boldsymbol{p}}_l)$ is the similarity term between Gabor features for the node $\hat{p}_l$ and p_l on the images I_l and M_l. $e_{p_l}^d(\hat{\boldsymbol{p}}_l)$ represents the elasticity of the graph on the image I_l. λ_d is a constant parameter for the graph elasticity. $\aleph_{p_l}$ is a set of nearest neighbor nodes p_l' for p_l. $D_{p_l,p_l'}^{M_l}$ and $D_{\hat{p}_l,p_l'}^{I_l}$ are the Euclidean distance between nodes p_l (or $\hat{p}_l$) and p_l' on the graph of the image M_l (or I_l). $A_{p_l,p_l'}^{M_l}$ and $A_{\hat{p}_l,p_l'}^{I_l}$ take a vector form consisting of 4 elements. One element represents an angular between two nearest neighbors on each quadrant, centered at p_l.

We realize that nodes are wandering more rapidly around the most appropriate position on the graph. For this, we employ the method of maximum operation. It is assumed that neurons in a layer compete with each others for activation. In our maximum operation method, a search region with $R = (2q + 1) \times (2q + 1)$ centered at each node p_l are prepared ($q = 4$). In search region, each node surveys to find its pixel position $\bar{\boldsymbol{p}}_l$ that the energy function $E_l(\hat{\boldsymbol{p}}_l)$ takes the maximum value:

$$\bar{\boldsymbol{p}}_l = \max_{\hat{\boldsymbol{p}}_l \in R} \{E_l(\hat{\boldsymbol{p}}_l)\}. \qquad (9)$$

3.4 Simulations

If each node on a model image detects the most likely pixel positions on the lowest resolution image I_4, the positions are normalized for the next higher
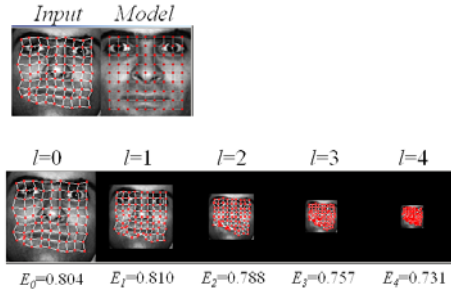
Fig. 4. Distortion invariance recognition is successfully achieved by Gabor filter construction model

Table 1. A recognition ability of the Gabor filter construction model by using 6 facial images of different persons (M). Subscripts of I mean a scale parameter controlling the size of an input image.

	I_0	I_1	I_2	I_3	I_4
M1	0.974	0.740	0.549	0.436	0.067
M2	0.982	0.733	0.637	0.403	0.047
M3	0.959	0.776	0.671	0.459	0.092
M4	0.976	0.774	0.650	0.413	0.231
M5	0.987	0.743	0.615	0.421	0.167
M6	1.000	0.756	0.703	0.441	0.404

resolution image I_3 to set up again with the search region. Then, it starts again to calculate E_3 for finding the most likely pixel position. It is repeatedly done until the input image with the original size I_0. This is shown in Fig.4. In Fig.4, distortion invariant recognition is carried out.

In Table 1, we can test a recognition ability of constructed Gabor filter model proposed in this work, using 6 facial images of different persons. When a model image $M6$ is used as an input image, this case gets the highest similarities, compared to the cases using the other model images as the input.

4   Discussion

One can already see that in a constructed Gabor feature model proposed here, there are still ample discussions about how to improve and develop the model toward practical application in the near future. For example, the constructed Gabor feature model possesses the important functionality of translation invariance in this work. In addition, another important functionality, namely, the size and rotation invariance will have to be employed for creating full invariant visual object recognition system. A Gabor feature decomposition algorithm, proposed

by Sato et al. [13] is a powerful and helpful tool to developmental full invariant recognition system. At this moment, we should keep in mind for comparing the SIFT feature detectors.

The main concept of the Gabor pyramid technique must be "finding scale feature correspondences." The scale feature correspondence finding provides us with fruitful advantages, mainly, computational cost reduction, which has been well-known in using Laplacian or Gaussian filters [11]. However, we have to be aware that memory size reduction must also be significant. This is because if we implement conventional EGM and EBGM onto computers, 8 different orientations and 5 different spatial frequencies of Gabor filters are necessary for one 256×256 pixel size image. We had to prepare $256 \times 256 \times 40 (= 2621440)$ memory size per one image on the database. However, if applying Gabor pyramid to EGM and EBGM, we use a down-sampling size image for each scale. Since total memory size is $(256 \times 256 + 180 \times 180 + 128 \times 128 + 90 \times 90 + 64 \times 64) \times 8$ $(= 1012128)$, we can remove size redundancy with around 38.6%.

The memory size reduction will have the greater value when Gabor pyramid technique is implemented into a digital hardware circuit of an FPGA. Because the conventional EGM utilizes 5 different spatial frequencies of the Gabor function, the FPGA circuit can easily be expected to need at least 5 blocks for a Gabor function. However, if taking into account the Gabor pyramid, we need only one block for a Gabor kernel with the standard level, plus to one logic circuit for down-sampling. Another merit is an expectation that the object detection accuracy may be better, compared to the case with no reduction memory size. In any case, we will have to discuss more Gabor pyramid implementation into FPGA, because the FPGA implementation has high potential to practical applications such as robot vision, environmental surveillance systems and so on. Also, it has great and innovative expectations of the real time robot vision.

5   Conclusion

In this paper, we propose a neurally inspired model for visual object recognition. This model is based on the Gabor feature explicit representation construction. An input image is decomposed of different scale features through the low-pass filter. Nevertheless, recycling and overlapping again the scale features, a model object can be detected on the input image by finding scale feature correspondence to the memory representations. Simultaneously, Gabor feature representations stored in memory are also reconstructed by selecting the most similar scale features to the input. Also, we discuss about how to implement our Gabor feature construction model into the FPGA, which will furthermore be implemented into practical application in the near future.

References

1. Lowe, D.G.: Object Recognition from Local Scale-Invariant Features. In: Proceedings of the International Conference on Computer Vision 1999, vol. 2, pp. 150–1157 (1999)
2. Viola, P., Jones, M.J.: Robust Real-Time Face Detection. International Journal of Computer Vision 57(2), 137–154 (2004)
3. Riesenhuber, M., Poggio, T.: Hierarchical Models of Object Recognition in Cortex. Nature Neuroscience 2, 1019–1025 (1999)
4. Fukushima, K.: Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics 36(4), 193–202 (1980)
5. von der Malsburg, C.: Vision as Dynamical System. Under submission (2011)
6. Lades, M., Vorbrueggen, J.C., Buhmann, J., et al.: Distortion invariant object recognition in the dynamic link architecture. IEEE Transactions on Computers 42(3), 300–311 (1993)
7. Wolfrum, P., Wolff, C., Lucke, J., von der Malsburg, C.: A Recurrent Dynamic Model for Correspondence-Based Face Recognition. Journal of Vision 8, 1–18 (2008)
8. Cadik, M.: Human Perception and Computer Graphics. Czech Technical University Postgraduate Study Report (2004)
9. da Fontoura Costa, L., Cesar, R.M.: Shape analysis and classification: theory and practice. CRC Press, USA (2001)
10. Thibos, L.N.: Image processing by the human eye. In: Pearlman, W.A. (ed.) Proc. SPIE, vol. 1199, pp. 1148–1153 (1989)
11. Fujiyoshi, H.: Gradient-Based Feature Extraction: SIFT and HOG. Technical report of IEICE. PRMU 107(206), 211–224 (2004)
12. Eleyan, A., Ozkaramanli, H., Demirel, H.: Complex wavelet transform-Based face recognition. EURASIPJournal on Advances in Signal Processing 2008, Article ID 185281, 13 pages (2008)
13. Sato, Y.D., Jitsev, J., von der Malsburg, C.: A Visual Object Recognition System Invariant to Scale and Rotation. In: Kůrková, V., Neruda, R., Koutník, J. (eds.) ICANN 2008, Part I. LNCS, vol. 5163, pp. 991–1000. Springer, Heidelberg (2008)

Multiview Range Image Registration
Using Competitive Associative Net
and Leave-One-Image-Out Cross-Validation Error

Shuichi Kurogi, Tomokazu Nagi, Shoichi Yoshinaga,
Hideaki Koya, and Takeshi Nishida

Kyusyu Institute of technology, Tobata, Kitakyushu, Fukuoka 804-8550, Japan
{kuro@,nagi@kurolab.,yoshinaga@kurolab.}cntl.kyutech.ac.jp
http://kurolab.cntl.kyutech.ac.jp/

Abstract. This paper presents a method for multiview range image registration
to fuse 3D surfaces in range images taken from around an object by a laser range
finder (LRF). The method uses competitive associative net (CAN2) for learning
piecewise linear approximation of surfaces in the LRF range image involving var-
ious noise, and then executes pairwise registration of consecutive range images
approximated by piecewise planes. To reduce the propagation error caused by
the consecutive pairwise registration, the method introduces leave-one-image-out
cross-validation (LOOCV) and tries to minimize the LOOCV registration error.
The effectiveness is shown by using real LRF range images of several objects.

Keywords: Multiview range image registration, LRF range images, Competitive
associative nets, Leave-one-image-out cross-validation.

1 Introduction

This paper describes multiview range image registration to fuse three-dimensional (3D)
surfaces in range images taken from around an object by the laser range finder (LRF).
Here, the LRF range images are characterized as involving lack of data called black
spots, quantization errors owing to the range (distance) resolution (e.g. 10mm), and a
large number of data owing to high angular resolution (e.g. $0.25°$). To deal with such
data, we have developed a plane extraction method using CAN2 (competitive associa-
tive net) [1], where the CAN2 is an artificial neural net for learning piecewise linear
approximation of nonlinear functions and provides several advantages such as data
compression, noise reduction, availability of approximated piecewise plane segments
for processing LRF range images. We have utilized these advantages in range image
registration [2] using particle filter (PF) and loop-closing for reducing the propagation
error caused by the pairwise registration applied to consecutive images. From the sur-
veys of range image registration [3,4], our method is classified into statistic techniques
for cycle minimization studied for SLAM (simultaneous localization and mapping)
problems [4], where, however, not the PF but some Gaussian filters are only mentioned
so that our method has to be analyzed from various points of view.

Although the above our method seems to work well, this paper presents a new mul-
tiview registration method using the CAN2 and leave-one-image-out cross-validation
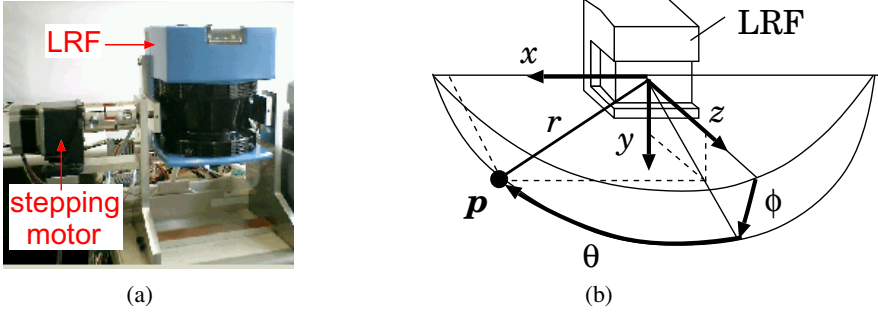
Fig. 1. (a) The LRF with the stepping motor and (b) the LRF coordinate system

(LOOCV) error. From the survey [4], the present method is closely related to the metaview strategies of analytic techniques for cycle minimization to reduce the propagation error. The conventional metaview methods incrementally register and merge the consecutive images into a metaview, while the present method generate all LOOCV metaviews, and tries to reduce the registration error between every pair of LOOCV metaview and the remaining image. Here, the correspondence of points on the LOOCV metaview and the remaining image is checked by means of the distance and the orientation of the tangent planes at the points, which is easy to be implemented with the range images represented by the piecewise planes obtained by the CAN2.

In the next section, we show the present method for multiview registration using CAN2 and LOOCV error, and then the effectiveness of the method is evaluated in Sect. 3.

2   Range Image Registration Using CAN2 and LOOCV Error

2.1   LRF Range Data

The SICK LMS200 is used as a LRF for scanning the horizontal 2D plane to measure the distance to an object and a suspension unit for rotating the LRF vertically by means of a geared stepping motor (see Fig. 1(a)). The yaw and pitch angle resolutions are $0.25°$ and $0.05°$, respectively, and the range (distance) resolution is 10mm. Let $Z_{[\text{polar}]t} = \{\boldsymbol{p}^{(i)}_{[\text{polar}]t} = (\theta^{(i)}_t, \phi^{(i)}_t, r^{(i)}_t) \mid i = 1, 2, \cdots\}$ denote the tth range image taken from around an object, where $\theta^{(i)}_t$, $\phi^{(i)}_t$ and $r^{(i)}_t$, respectively, are the yaw and pitch angles and the range of the ith scan data (see Fig. 1(b)), and $t \in I^{\text{img}} = \{1, 2, \cdots, L\}$. From $Z_{[\text{polar}]t}$, we have the Cartesian data as $\boldsymbol{p}^{(i)}_{[\text{s}]t} = (x^{(i)}_{[\text{s}]t}, y^{(i)}_{[\text{s}]t}, z^{(i)}_{[\text{s}]t})^T$ $= r^{(i)}_t (\sin\theta^{(i)}_t, \cos\theta^{(i)}_t \sin\phi^{(i)}_t, \cos\theta^{(i)}_t \cos\phi^{(i)}_t)^T$, where the subscript [s] indicates the scan center coordinate system because the z-axis of $\boldsymbol{p}^{(i)}_{[\text{s}]t}$ directs to the center of the LRF scan. We denote this dataset as $Z_{[\text{s}]t} = \{\boldsymbol{p}^{(i)}_{[\text{s}]t} = (x^{(i)}_{[\text{s}]t}, y^{(i)}_{[\text{s}]t}, z^{(i)}_{[\text{s}]t})^T \mid i = 1, 2, \cdots\}$.

The registration from the tth image to the $(t-1)$th image is executed by the transformation $\boldsymbol{p}^{(i)}_{[\text{s}]t-1,t} = \boldsymbol{R}_{[\text{s}]t-1,t}\boldsymbol{p}^{(i)}_{[\text{s}]t} + \boldsymbol{t}_{[\text{s}]t-1,t}$, where the parameter $u_t = (\boldsymbol{R}_{[\text{s}]t-1,t}, \boldsymbol{t}_{[\text{s}]t-1,t})$ consists of the rotation matrix $\boldsymbol{R}_{[\text{s}]t-1,t}$ and the translation

vector $\boldsymbol{t}_{[s]t-1,t}$. By means of applying this relation recursively, we can transform $\boldsymbol{p}_{[s]t}^{(i)}$ to the 1st LRF coordinate system as $\boldsymbol{p}_{[s]1,t}^{(i)} = \boldsymbol{R}_{[s]1,t}\boldsymbol{p}_{[s]t}^{(i)} + \boldsymbol{t}_{[s]1,t}$, where

$$x_t = (\boldsymbol{R}_{[s]1,t}, \boldsymbol{t}_{[s]1,t}) = (\boldsymbol{R}_{[s]1,t-1}\boldsymbol{R}_{[s]t-1,t}, \boldsymbol{t}_{[s]1,t-1} + \boldsymbol{R}_{[s]1,t-1}\boldsymbol{t}_{[s]t-1,t}). \quad (1)$$

indicates the pose, or the orientation $\boldsymbol{R}_{[s]1,t}$ and the position $\boldsymbol{t}_{[s]1,t}$, of the tth LRF w.r.t. the 1st LRF coordinate system, and u_t indicates the movement of the LRF.

2.2 Piecewise Planes Approximated by the CAN2

In order to utilize a mapping from (x, y) position to z position, we rotate $\boldsymbol{p}_{[s]t}^{(i)}$ to

$$\boldsymbol{p}_{[o]t}^{(i)} \triangleq \boldsymbol{R}_{[o,s]t}\boldsymbol{p}_{[s]t}^{(i)} \equiv \boldsymbol{R}_P(-\phi_t^{(0)})\boldsymbol{R}_Y(-\theta_t^{(0)})\boldsymbol{p}_{[s]t}^{(i)}, \quad (2)$$

on the coordinate system whose z-axis directs to the object center, where $\boldsymbol{R}_Y(-\theta_t^{(0)})$ and $\boldsymbol{R}_P(-\phi_t^{(0)})$ denote the yaw and the pitch rotation matrices.

We use CAN2 with N units for each tth range image. The jth unit has a weight vector $\boldsymbol{w}_{[o]t}^{(j)} = (w_{[o]t,1}^{(j)}, w_{[o]t,2}^{(j)})^T$ and an associative matrix (row vector) $\boldsymbol{M}_{[o]t}^{(j)} = (M_{[o]t,0}^{(j)}, M_{[o]t,1}^{(j)}, M_{[o]t,2}^{(j)})$. After learning $Z_{[o]t} = \{\boldsymbol{p}_{[o]t}^{(i)} = (x_{[o]t}^{(i)}, y_{[o]t}^{(i)}, z_{[o]t}^{(i)})^T | i = 1, 2, \cdots \}$ as a function $y = f(\boldsymbol{x})$ for $\boldsymbol{x} = (x_{[o]t}^{(i)}, y_{[o]t}^{(i)})^T$ and $y = z_{[o]t}^{(i)}$, the CAN2 divide the input space into Voronoi regions $V_{[o]t}^{(j)} = \{\boldsymbol{x} | j = \operatorname*{argmin}_i \{\|\boldsymbol{x} - \boldsymbol{w}_{[o]t}^{(i)}\|\} \text{ for } j \in I^{\mathrm{CAN2}} = \{1, 2, \cdots, N\}$, and performs linear approximation $y = \boldsymbol{M}_{[o]t}^{(j)}\boldsymbol{x}$ in each region. As a result, the range image is divided into piecewise planes given by $(\boldsymbol{n}_{[o]t}^{(j)})^T\boldsymbol{P}_{[o]t} = \alpha_{[o]t}^{(j)}$, where the normal vector $\boldsymbol{n}_{[o]t}^{(j)} = (n_{[o]t,x}^{(j)}, n_{[o]t,y}^{(j)}, n_{[o]t,z}^{(j)})^T$ and the distance to the origin $\alpha_{[o]t}^{(j)}$ are given by

$$((\boldsymbol{n}_{[o]t}^{(j)})^T, \alpha_{[o]t}^{(j)}) = \frac{(-M_{[o]t,1}^{(j)}, -M_{[o]t,2}^{(j)}, 1, M_{[o]t,0}^{(j)})}{\sqrt{(M_{[o]t,1}^{(j)})^2 + (M_{[o]t,2}^{(j)})^2 + 1}}. \quad (3)$$

Here, note that $n_{[o]t,z}^{(j)} > 0$ or the normal vector directs forward from the tth LRF. We use $Z_{[o]t}^{\mathrm{CAN2}} = \{\boldsymbol{q}_{[o]t}^{(j)} = (\boldsymbol{w}_{[o]t}^{(j)}, \boldsymbol{M}_{[o]t}^{(j)}, \widetilde{\boldsymbol{w}}_{[o]t}^{(j)})^T | j \in I^{\mathrm{CAN2}}\}$ for registration as follows.

ROI for Registration: From $Z_{[o]t}^{\mathrm{CAN2}}$, we remove the following data and obtain the ROI (Region of Interest) dataset $Z_{[o]t}^{\mathrm{ROI}} = \{\boldsymbol{q}_{[o]t}^{(j)} | j \in I_{[o]t}^{\mathrm{ROI}}\}$.

(i) (Remove floor) By means of the plane extraction method using the CAN2 [5], we extract the floor plane from $Z_{[o]t}^{\mathrm{CAN2}}$, and remove the data within the distance $d^{\mathrm{floor}}(= 30\mathrm{mm})$ to the floor.

(ii) (Remove jump edge) The data on the jump edge hold $(\boldsymbol{n}_{[\text{o}]t}^{(j)})^T \boldsymbol{q}_{[\text{o}]t}^{(j)} = 0$. So, we remove the data with $|(\boldsymbol{n}_{[\text{o}]t}^{(j)})^T \boldsymbol{q}_{[\text{o}]t}^{(j)}|/\|\boldsymbol{q}_{[\text{o}]t}^{(j)}\| < \cos(\pi/2 - \psi^{\text{je}})$, where $\psi^{\text{je}}(= 5°)$ indicates allowable error.

(iii) (Remove unreliable piecewise planes) We remove the data in the Voronoi region of the unit which involves less than 4 data because the plane is unreliable.

Let us consider the registration of the cth image $Z_{[\text{o}]c}^{\text{ROI}}$ to the $r(= c - 1)$th image $Z_{[\text{o}]r}^{\text{ROI}}$, where c and r represent the current and reference images, respectively.

2.3   Pairwise Registration Using Planes Extracted by the CAN2

Suppose the plane extraction method [5] extracts the centers $\boldsymbol{\zeta}_{[\text{o}]c}^{(j)}$ and the normal vectors $\boldsymbol{\nu}_{[\text{o}]c}^{(j)}$ of plane surfaces on the object for $c \in I^{\text{img}}$, and there is a common plane surface on both cth and $r(= c - 1)$th images. Then, an adequate transformation for registration is supposed to be obtained by the following steps, where let us suppose that the jth plane in the cth image corresponds to the s_jth plane in the rth image.

[Algorithm: Pairwise Registration]

Step 1. (Coarse registration) For each correspondence s_j, we obtain the yaw angle of corresponding planes by $\theta_{[\text{s}]r,c}^{(s_j)} := \text{atan2}(\nu_{[\text{s}]r,z}^{(s_j)}, \nu_{[\text{s}]r,x}^{(s_j)}) - \text{atan2}(\nu_{[\text{s}]c,z}^{(j)}, \nu_{[\text{s}]c,x}^{(j)})$. where $\boldsymbol{\nu}_{[\text{s}]t}^{(j)} = (\nu_{[\text{s}]t,x}^{(j)}, \nu_{[\text{s}]t,y}^{(j)}, \nu_{[\text{s}]t,z}^{(j)})$ is the normal vector represented by the scan center coordinate, and $\text{atan2}(z, x)$ gives the angle of the point (x, z) from the positive x-axis. Then, the transformation of the points in the cth image to the rth image, $\boldsymbol{q}_{[\text{o}]r,c}^{(j)} := \boldsymbol{R}_{[\text{o}]r,c} \boldsymbol{q}_{[\text{o}]c}^{(j)} + \boldsymbol{t}_{[\text{o}]r,c}$, is obtained, where $(\boldsymbol{R}_{[\text{o}]r,c}, \ \boldsymbol{t}_{[\text{o}]r,c}) := (\boldsymbol{R}_{[\text{o},\text{s}]r} \boldsymbol{R}_Y(\theta_{[\text{s}]r,c}^{(s_j)}), \boldsymbol{\zeta}_{[\text{o}]r}^{(s_j)} - \boldsymbol{R}_{[\text{o}]r,c} \boldsymbol{\zeta}_{[\text{o}]c}^{(j)})$.

Step 2. (Obtain ROI) We transform the cth data points $\boldsymbol{q}_{[\text{o}]c}^{(j)} \in Z_{[\text{o}]c}^{\text{ROI}}$ to the rth coordinate system as $\boldsymbol{q}_{[\text{o}]r,c}^{(j)} := \boldsymbol{R}_{[\text{o}]r,c} \boldsymbol{q}_{[\text{o}]c}^{(j)} + \boldsymbol{t}_{[\text{o}]r,c}$. and let $\boldsymbol{q}_{[\text{s}]r}^{(l_j)}$ be the closest point on the rth image, and $Z_{[\text{o}]r,c}^{\text{ROI}} := \{\boldsymbol{q}_{[\text{o}]r,c}^{(j)} | j \in I_{r,c}^{\text{ROI}}\}$ be the set of $\boldsymbol{q}_{[\text{o}]r,c}^{(j)}$ holding $\|\boldsymbol{q}_{[\text{s}]r,c}^{(j)} - \boldsymbol{q}_{[\text{s}]r}^{(l_j)}\| \leq d^{\text{ROI}}$ and $\left(\boldsymbol{n}_{[\text{s}]r,c}^{(j)}\right)^T \boldsymbol{n}_{[\text{s}]r}^{(l_j)} \geq \cos \psi^{\text{ROI}}$, where d^{ROI} and ψ^{ROI} are thresholds.

Step 3. (Evaluate the registration) We obtain the squared error given by

$$(\Delta Z_{r,c})^2 = \left\langle \|\boldsymbol{q}_{[\text{o}]r,c}^{(j)} - \boldsymbol{q}_{[\text{o}]r}^{(l_j)}\|^2 \right\rangle_{\boldsymbol{q}_{[\text{o}]r,c}^{(j)} \in Z_{[\text{o}]r,c}^{\text{ROI}}}. \tag{4}$$

Here, the angle brackets $\langle \cdot \rangle$ indicate the mean and the subscript indicates the range of the mean. After obtaining $(\Delta Z_{r,c})^2$ for all correspondences s_j, let $(\widehat{\boldsymbol{R}}_{[\text{o}]r,c}, \widehat{\boldsymbol{t}}_{[\text{o}]r,c})$ be the transformation which achieves the smallest $(\Delta Z_{r,c})^2$.

The result of the above (coarse) pairwise registration is fed to the multiview registration to be refined as shown in the next section.

2.4   Multiview Registration Using LOOCV Error

From $\left(\widehat{R}_{[\mathrm{o}]r,c}, \widehat{t}_{[\mathrm{o}]r,c}\right)$ via Eq.(1) and (2), all LRF poses $\left(\widehat{R}_{[\mathrm{s}]1,c}, \widehat{t}_{[\mathrm{s}]1,c}\right)$ on the scan center coordinate system of the 1st LRF are derived for $c = 2, 3, \cdots, L$, consecutively. However, they involve propagation error, and we employ the following method to reduce the LOOCV registration error.

[Algorithm: Multiview Registration]

Step 1. (Coarse Registration) Let $(R_{[\mathrm{s}]1,c}, t_{[\mathrm{s}]1,c}) := (\widehat{R}_{[\mathrm{s}]1,c}, \widehat{t}_{[\mathrm{s}]1,c})$ for all $c \in I^{\mathrm{img}}$.

Step 2. (Obtain ROI) We obtain $q_{[\mathrm{s}]1,c}^{(j)} := R_{[\mathrm{s}]1,c} q_{[\mathrm{s}]c}^{(j)} + t_{[\mathrm{s}]1,c}$ for all $j \in I_{[\mathrm{o}]c}^{\mathrm{ROI}}$ in the cth image, and let $q_{[\mathrm{s}]1,\bar{c}}^{(l_j)}$ be the closest point for all $\bar{c} \in I^{\mathrm{img}} \backslash \{c\}$ and $Z_{[\mathrm{s}]1,c}^{\mathrm{ROI}} := \{q_{[\mathrm{s}]1,c}^{(j)} | j \in I_{1,c}^{\mathrm{ROI}}\}$ be the set of $q_{[\mathrm{s}]1,c}^{(j)}$ holding $\|q_{[\mathrm{s}]1,c}^{(j)} - q_{[\mathrm{s}]1,\bar{c}}^{(l_j)}\| \leq d^{\mathrm{ROI}}$ and $\left(n_{[\mathrm{s}]1,c}^{(j)}\right)^T n_{[\mathrm{s}]1,\bar{c}}^{(l_j)} \geq \cos \psi^{\mathrm{ROI}}$ for the thresholds d^{ROI} and ψ^{ROI}.

Step 3. (Refine transformation) In order to improve the accuracy, we obtain the point $\xi_{[\mathrm{s}]1,\bar{c}}^{(l_j)} := q_{[\mathrm{s}]1,\bar{c}}^{(l_j)} + \left(\alpha_{[\mathrm{s}]1,\bar{c}}^{(l_j)} - (n_{[\mathrm{s}]1,\bar{c}}^{(l_j)})^T q_{[\mathrm{s}]1,c}^{(j)}\right) n_{[\mathrm{s}]1,\bar{c}}^{(l_j)}$ which is on the tangent plane of $q_{[\mathrm{s}]1,\bar{c}}^{(l_j)}$ and closest to $q_{[\mathrm{s}]1,c}^{(j)}$, and introduce

$$\eta_{[\mathrm{s}]1,\bar{c}}^{(l_j)} := \begin{cases} \xi_{[\mathrm{s}]1,\bar{c}}^{(l_j)} & \text{if } \|q_{[\mathrm{s}]1,c}^{(j)} - q_{[\mathrm{s}]1,\bar{c}}^{(l_j)}\| \leq d^{\mathrm{p2pl}}, \\ q_{[\mathrm{s}]1,\bar{c}}^{(l_j)} & \text{otherwise,} \end{cases} \tag{5}$$

where d^{p2pl} indicates a threshold to use point-to-plane distance. Now, we introduce the registration error given by

$$\left\langle (\Delta Z_{c,\bar{c}})^2 \right\rangle_{c \in I^{\mathrm{img}}} = \left\langle \|\Delta R_c\, q_{[\mathrm{s}]1,c}^{(j)} + \Delta t_c - \eta_{[\mathrm{s}]1,\bar{c}}^{(l_j)}\|^2 \right\rangle_{q_{[\mathrm{s}]1,c}^{(j)} \in Z_{[\mathrm{s}]1,c}^{\mathrm{ROI}}, c \in I^{\mathrm{img}}}. \tag{6}$$

This indicates the LOOCV MSE (mean-square-error) of the current registration with $(\Delta R_c, \Delta t_c) = (I, 0)$. From the ICP (iterative-closest-point) methods, the error is expected to be reduced by $(\Delta R_c, \Delta t_c) := \left(U_c V_c^T, \overline{\eta_{[\mathrm{s}]1,\bar{c}}^{(l_j)}} - \Delta R_c \overline{q_{[\mathrm{s}]1,c}^{(j)}}\right)$, where U_c and V_c are the left and the right singular matrices of the cross-covariance matrix of $\eta_{[\mathrm{s}]1,\bar{c}}^{(l_j)}$ and $q_{[\mathrm{s}]1,c}^{(j)}$ with the mean vectors $\overline{\eta_{[\mathrm{s}]1,\bar{c}}^{(l_j)}}$ and $\overline{q_{[\mathrm{s}]1,c}^{(j)}}$, respectively. We update the transformation as $(R_{[\mathrm{s}]1,c}, t_{[\mathrm{s}]1,c}^{(j)}) := (\Delta R_c R_{[\mathrm{s}]1,c}, \Delta R_c t_{[\mathrm{s}]1,c}^{(j)} + \Delta t_c)$ for all c and repeat **Step 2** and **3** until convergence.

3   Experimental Results and Remarks

In order to examine the effectiveness of the present method, we have first conducted an experiment using four range images taken from around a rectangular box on the floor. The result is shown in Fig. 2, where we use the 1st image as the 5th image to examine the propagation error.   From (a), (b) and (c), we can see the performance of the pairwise registration. Namely, from (a), we can see the propagation error as the
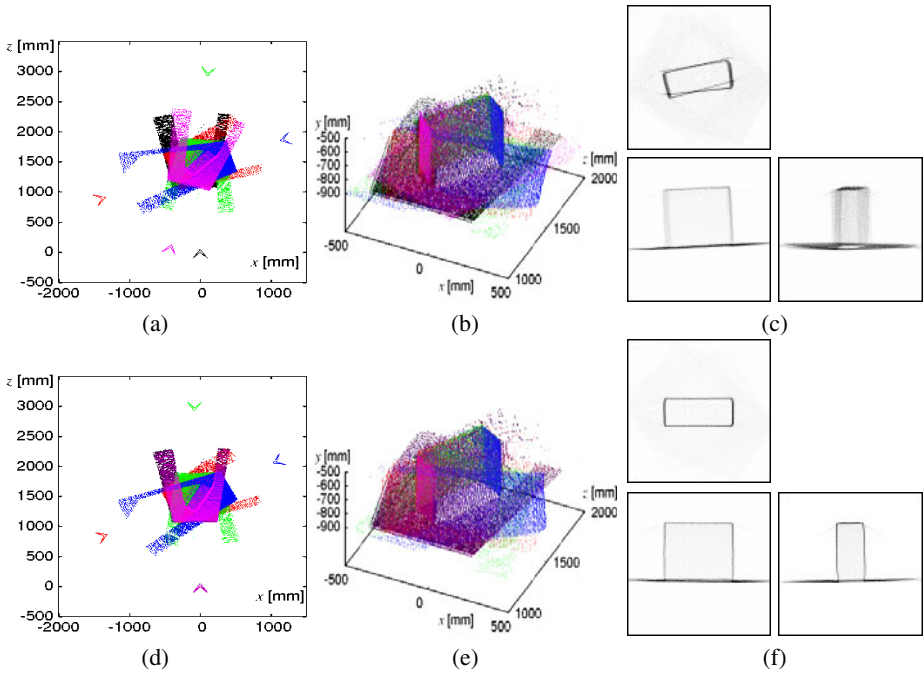
Fig. 2. Experimental result using four range images taken from around a rectangular box. We use the CAN2 with $N = 2000$. (a),(b) and (c) indicate the result of the pairwise registration. (a) shows the over-head view, where the arc-heads colored black, red, green, blue and pink indicate the registered poses of the tth LRF for $t = 1, 2, 3, 4, 5$, respectively, and the dots indicate the LRF data points, where we use the 1st image as the 5th image to examine the propagation error. (b) shows the perspective view. (c) shows the top, front and side view of the registered object, where the range data are digitized into $10mm^3$ cubic volumes in the $1m^3$ cubic area involving the object, and the numbers of the data in the volumes are summed up orthogonally to each view, respectively. (d), (e) and (f) show the result of multiview registration.

difference of the 1st and 5th LRF poses, or the black and the pink arc-heads. We can also see the propagation error in the top view of (c). From (d) obtained by the multiview registration, we can see that the pink poses for $t = 5$ goes to almost at the origin $(x, y, z) = (0, 0, 0)$, which indicates the propagation error is reduced. The reduction of the propagation error can be seen in (f).

We have conducted experiments for two other objects, which we call box-on-box and doll-on-box as shown in Fig. 3. We can see that the propagation error of the pairwise registration is reduced by the multiview registration.

From the thick edges of the three side views, we have estimated the size of boxes and show the result in Table 1. Considering that every surface of the boxes has 10mm quantization error, we can say that the estimated values are almost accurate.

In Fig. 4(a), we show the LOOCV RMSE vs. the number of iterations. We can see the reduction of the registration error by means of the point-to-plane technique. Note that the RMSE is not as small as the one (below 5mm) shown in [3]. It is supposed to be owing that our data involve 10mm quantization error of the LRF. Furthermore,
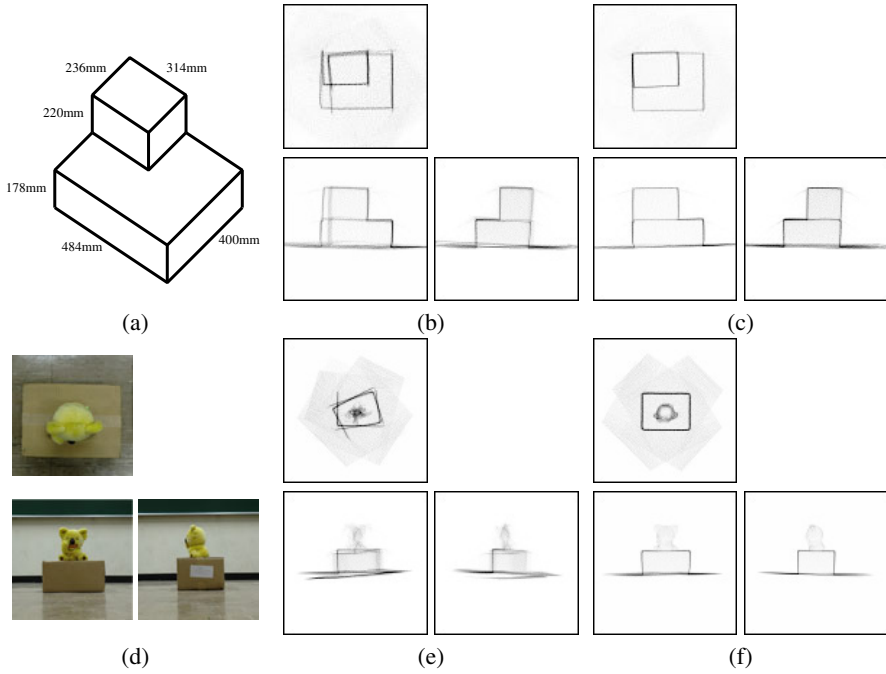
Fig. 3. Experimental results for box-on-box object ((a), (b), (c)) and doll-on-box object ((d),(e),(f)). The results of pairwise registration are shown in (b) and (e), and those of multiview registration are shown in (c) and (f).

Table 1. Actual and estimated size (Depth, Width and Height) of boxes. The unit of actual and estimated sizes are [mm] and [pixel/10mm], respectively.

	box			box-on-box						doll-on-box		
box size	D	W	H	D	W	H	D	W	H	D	W	H
actual	175	485	396	236	314	220	400	484	178	238	322	161
estimated	18-20	45-46	40-41	23-24	32-33	22-23	40-41	49-50	19-20	25-26	34-35	17-18

our multiview registration has failed with $d^{\mathrm{p2pl}} = d^{\mathrm{ROI}}\ (= 50\mathrm{mm})$ for box-on-box and doll-on-box objects, which is also supposed to be owing to the quantization error. Actually, in order to overcome this problem, we have introduced d^{p2pl} smaller than d^{ROI} to use point-to-plane distance for accuracy and point-to-point for stability.

In Fig. 4(b) and (c), we show some results related to the CAN2, where the original range image data points $\boldsymbol{p}_{[\mathrm{o}]t}^{(i)}$ are compressed into the points $\boldsymbol{q}_{[\mathrm{o}]t}^{(j)}$ by the batch learning method of the CAN2 [6] and they are classified into the points on several planes by the plane extraction method [1], and then they are used for obtaining the ROI and the registration.

The LOOCV is known to require a large computational cost in general, but in this application, most of the computational cost is used for the learning of the CAN2 to learn piecewise linear approximation of the range image, and the computational cost of the multiview iterations vs. the total cost is 20s/408s, 27s/380s and 36s/653s for box, box-on-box, and doll-on-box objects, respectively.
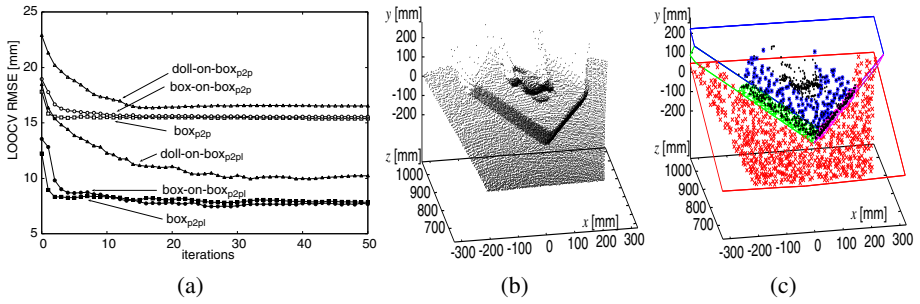
Fig. 4. Performance of registration. (a) The LOOCV RMSE vs. the number of iterations through multiview registration. The LOOCV RMSE is obtained by the root of the LOOCV MSE given by Eq.(6), where $d^{\mathrm{p2pl}} = 0$ for the point-to-point only (p2p) and $d^{\mathrm{p2pl}} = 40$mm for the point-to-plane (p2pl) in Eq.(5), $d^{\mathrm{ROI}} = 50$mm and $\psi^{\mathrm{ROI}} = 25°$. (b) shows an example of the original 15027 range data points $\boldsymbol{p}_{[\mathrm{o}]t}^{(i)}$ of the doll-on-box object, and (c) depicts the points $\boldsymbol{q}_{[\mathrm{o}]t}^{(j)}$ on the 4 extracted planes (1112 colored points) and the points in $Z_{[\mathrm{o}]t}^{\mathrm{ROI}}$ (773 black points).

The present LOOCV based multiview registration is supposed to be possible owing mainly that **Step 2** of the multiview registration algorithm utilizes the orientation threshold ψ^{ROI} as well as the distance threshold d^{ROI} to check the point to be corresponded, while the most of the conventional registration methods seem to use only the distance threshold.

4   Conclusion

We have presented a multiview registration method using CAN2 and LOOCV registration error to fuse 3D surfaces of range images taken from around an object by the LRF. The effectiveness is shown and examined through the experimental results with real LRF range images. We would like to evaluate the present method much more on accuracy, robustness, and other performances compared with other methods.

References

1. Kurogi, S., Koya, H., Nagashima, R., Wakeyama, D., Nishida, T.: Range image registration using plane extraction by the CAN2. In: Proc. of CIRA (2009)
2. Kurogi, S., Nagi, T., Nishida, T.: Range image registration using particle filter and competitive associative nets. In: Wong, K.W., Mendis, B.S.U., Bouzerdoum, A. (eds.) ICONIP 2010, Part II. LNCS, vol. 6444, pp. 352–359. Springer, Heidelberg (2010)
3. Salvi, J., Matabosh, C., Foli, D., Forest, J.: A review of recent range image registration methods with accuracy evaluation. Image and Vision Computing 25, 578–596 (2007)
4. Batlle, E., Matabosch, C., Salvi, J.: Summarizing image/surface registration for 6dof robot/camera pose estimation. In: Martí, J., Benedí, J.M., Mendoncca, A.M., Serrat, J. (eds.) IbPRIA 2007. LNCS, vol. 4478, pp. 105–112. Springer, Heidelberg (2007)
5. Kurogi, S.: Plane extraction from range data using competitive associative nets. The Brain and Neural Networks 14(4), 273–281 (2007)
6. Kurogi, S., Sawa, M., Ueno, T., Fuchikawa, Y.: A batch learning method for competitive associative net and its application to function approximation. In: Proc. of SCI 2004, vol. 5, pp. 24–28 (2004)

Multi-view Pedestrian Recognition Using Shared Dictionary Learning with Group Sparsity

Shuai Zheng[1], Bo Xie[2], Kaiqi Huang[1], and Dacheng Tao[3]

[1] National Laboratory of Pattern Recognition,
Institute of Automation, Chinese Academy of Sciences, Beijing, China
{szheng,kqhuang}@nlpr.ia.ac.cn
[2] Chinese University of Hong Kong, Hongkong, China
zixu1986@gmail.com
[3] Centre for Quantum Computation and Intelligent Systems,
Faculty of Engineering and Information Technology,
University of Technology, Sydney, Australia
dacheng.tao@uts.edu.au

Abstract. Pedestrian tracking in multi-camera is an important task in intelligent visual surveillance system, but it suffers from the problem of large appearance variations of the same person under different cameras. Inspired by the success of existing view transformation model in multi-view gait recognition, we present a novel view transformation model based approach named shared dictionary learning with group sparsity to address the problem. It projects the pedestrian appearance feature descriptor in probe view into the gallery one before feature descriptors matching. In this case, $L_{1,\infty}$ regularization over the latent embedding ensure the lower reconstruction error and more stable feature descriptors generation, comparing with the existing Singular Value Decomposition. Although the overall optimization function is not global convex, the Nesterovs optimal gradient scheme ensure the efficiency and reliability. Experiments on VIPeR dataset show that our approach reaches the state-of-the-art performance.

Keywords: multiview learning, dimension reduction, stochastic neighbor embedding, image retrieval.

1 Introduction

Recognizing humans at a distance is not only a popular topic in computer vision community, but also very important in intelligent visual surveillance applications. However, this task is very challenging since there are tremendous variations in appearance and structure under different illumination, poses, occlusions and viewpoints [1,2,3]. The complex intra-class variations are difficult to model and lead to recognition failures. Besides, in uncontrolled scenarios, people may carry bags or exhibit other unexpected noises, which further deteriorate performance.

In this paper, we study human recognition in the multi-view setting. It refers to identifying a person across different viewpoints. This problem arises in numerous computer vision scenarios, such as pedestrian re-identification, multi-camera tracking, etc. The major issue is how to learn a view invariant representation while preserving discriminative information for accurate recognition. One paradigm is to use features such as Self
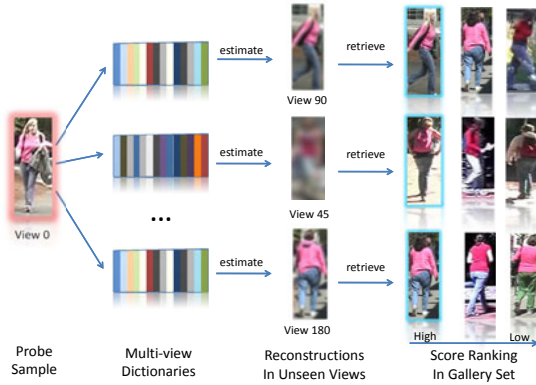
Fig. 1. Illustration of the multi-view human recognition system. Learned SDL-GS model is used to estimate data in unseen views given probe samples. In each unobserved view, the Nearest Neighbor (NN) rule is used as a classifier to retrieve samples in the gallery set.

Similarity Plots (SSP) [4] and flow fields gait representation [5]. Color histograms and texture filters are used in [6,7,5] to recognize pedestrian across different views. However, much discriminative information is lost in the process to achieve viewpoint invariance. For controlled scenes, viewpoint differences can be handled by view synthesis using planar imaging geometry or self calibration [8]. But this requires explicit knowledge of the scenes, which is unknown in most practical cases. Dikmen [7] proposed to learn a suitable metric where variations caused by view changes are offset by supervised learning. While label information is very important for recognition [5], neglecting the relationships between views is prone to sub-optimal solutions since conflicting views may be forced together simply because they are labeled as similar. Moreover, View Transformation Models (VTM) based on linear factorization [9,10] and Gaussian processes [8] also demonstrate satisfactory performance in multi-view gait recognition. However, these techniques are sensitive to noise and are likely to overfit in cases of small training sets.

When inferring the sparse latent embedding, traditional sparse coding techniques suffer from random locations of non-zero coefficients. We propose to use $l_{1,\infty}$ regularization on the latent embedding for group sparsity effect. Instead of being randomly distributed, the locations of non-zero coefficients will tend to cluster into groups. Such effect is resistant to noise disturbance and leads to improved recognition performance. Moreover, learning and inference in our approach only consists of convex optimization, which can be efficiently solved by Nesterov's optimal gradient scheme. Fig. 1 illustrates the mechanism of our approach at recognition time.

The contributions of our work are three-fold: 1) SDL-GS is capable of accurately estimating test samples in unseen views, so as to handle large variations; 2) Group sparsity leads to stable latent embedding and increases recognition performance significantly; 3) The computation of SDL-GS only consists of convex optimization and can be carried out efficiently with global optimum guarantee.

The remainder of the paper is organized as follows. In Section 2, we present our multi-view dictionary learning formulations. Group sparsity with $l_{1,\infty}$ norm regularization is introduced in Section 3. In Section 4, we evaluate our approach on VIPeR [6]. Finally, we conclude the paper in Section 5.

2 Shared Dictionary Learning with Group Sparsity

In this section, we first briefly review sparse coding techniques. Then we introduce our dictionary learning algorithm for problems with multiple views.

2.1 Sparse Coding

Recently, sparse coding or sparse representation is very popular within the face and object recognition communities[11]. It is reported that the reconstruction error of sparse coding is much lower than that of Singular Value Decomposition (SVD). Besides, sparse coefficients can be less sensitive to noise and occlusion [12]. But sparse coding has its drawback that the positions of non-zero coefficients in descriptor vectors are not stable, which leads it may not be robust enough.

Specifically, let $\mathbf{x}_i \in \mathbb{R}^d$ denote the i-th input data, which can be vectorized image pixels or extracted features. And denote $\mathbf{D} \in \mathbb{R}^{d \times k}$ as the dictionary matrix. We seek a sparse latent embedding to represent $\mathbf{x}_i$,

$$\boldsymbol{\alpha}_i = \arg\min_{\boldsymbol{\alpha}} \|\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}\|^2 + \lambda \|\boldsymbol{\alpha}\|_1 \tag{1}$$

where $\|\boldsymbol{\alpha}\|_1 = \sum_{j=1}^{k} |\alpha_j|$ is the l_1 norm and λ is the coefficient that balances the trade-off between the reconstruction error term and the regularization term.

2.2 Multi-view Dictionary Learning with Shared Latent Embedding

In our multi-view dictionary learning approach, we assume the sparse latent embedding is shared across all views, and each view-specific dictionary is responsible for generating the corresponding data samples.

This generative model provides a straightforward way of estimating unseen views given a subset of observed views. And our sparse dictionary-based approach can significantly reduce reconstruction error on unseen views and improve recognition accuracy compared with simple factorization approaches such as Singular Value Decomposition [9].

In details, given N data, each with V views, the goal is to learn V view-specific dictionaries that fit the data well. Let $\mathbf{x}_i^v$ denote the i-th data in the v-th view and $\mathbf{D}^v$ denote the dictionary for the v-th view. The dictionary learning approach is formulated as the following problem

$$\min_{\mathbf{D}^v, \boldsymbol{\alpha}_i} \frac{1}{V} \sum_{v=1}^{V} \sum_{i=1}^{N} \|\mathbf{x}_i^v - \mathbf{D}^v \boldsymbol{\alpha}_i\|^2 + \lambda \sum_{i=1}^{N} \|\boldsymbol{\alpha}_i\|_1 \tag{2}$$

$$s.t. \|\mathbf{d}_j^v\| \leq 1, \, 1 \leq j \leq k$$

where $\mathbf{d}_j^v$ is the j-th column of $\mathbf{D}^v$. Note that $\boldsymbol{\alpha}_i$ is shared across all views.

We denote the approach characterized in Eq. 3 as Shared Dictionary Learning with Sparse Coding (SDL-SC). SDL-SC is solved by alternating optimization. Fixing $\boldsymbol{\alpha}_i$'s, solving $\mathbf{D}^v$ is a quadratic programming problem and can be efficiently solved by Block Coordinate Descent [13]. Given $\mathbf{D}^v$, solving $\boldsymbol{\alpha}_i$'s is traditional sparse coding and can be efficiently carried out by approaches like LARS/Lasso [14] or feature-sign [15].

2.3   Inference on the Latent Embedding

Once we have learnt the set of dictionaries $\{\mathbf{D}^v\}_{v=1}^V$, we infer the shared latent embedding for a new test sample $\hat{\mathbf{x}}_i^v$ by solving the following problem

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\hat{\boldsymbol{\alpha}}} \frac{1}{V} \sum_{v=1}^V \|\hat{\mathbf{x}}_i^v - \mathbf{D}^v\hat{\boldsymbol{\alpha}}\|^2 + \lambda\|\hat{\boldsymbol{\alpha}}\|_1 \tag{3}$$

In some scenarios, only a subset of views $\mathcal{V}$ are available and we would like to reconstruct other views from the given data. This can be done by first inferring the latent embedding $\tilde{\boldsymbol{\alpha}}_i$ from the given views $\hat{\mathbf{x}}_i^v$, $v \in \mathcal{V}$

$$\min_{\tilde{\boldsymbol{\alpha}}_i} \frac{1}{|\mathcal{V}|} \sum_{v\in\mathcal{V}} \|\hat{\mathbf{x}}_i^v - \mathbf{D}^v\tilde{\boldsymbol{\alpha}}_i\|^2 + \lambda\|\tilde{\boldsymbol{\alpha}}_i\|_1 \tag{4}$$

where $|\mathcal{V}|$ is the cardinality of the set $\mathcal{V}$.

Then data in the unseen views can be estimated through $\tilde{\mathbf{x}}_i^v = \mathbf{D}^v\tilde{\boldsymbol{\alpha}}_i$, where $v \notin \mathcal{V}$.

3   Group Sparsity Induced by $l_{1,\infty}$ Norm

One problem with sparse coding is that the locations of non-zero coefficients of latent embedding are sensitive to noise – two similar data samples may have quite different sparse coding coefficients, thus making the sparse coding scheme unstable for similarity measure. Recently, Huang [16] proposed a dynamic group sparsity algorithm that can incorporate the prior of clustering trend of non-zero elements. Their approach is a greedy sparse recovery algorithm, with iterative pruning.

We introduce a group sparsity formulation with $l_{1,\infty}$ norm regularization. Compared with the pruning approach, our convex alternative is much simpler to implement and can take advantage of any existing convex optimization solvers. More importantly, our approach does not require any knowledge about the locations or sizes of the group clusters and can achieve superior results than traditional sparse coding techniques in almost all tasks.

Our formulation is similar to group sparse coding [17]. However, our approach adopts $l_{1,\infty}$ norm regularization, which is reported to exhibit better performance than $l_{1,2}$ norm [18]. Jia [19] also proposed a factorized multi-modal learning framework that places group sparsity constraints on the dictionaries. While these constraints are sensible on multi-modal data where data from different modalities are drastically different, we have observed better performance by simple norm constraints for multi-view data where different views have much in common.

3.1   Convex Group Sparsity

In order to faciliate the use of $l_{1,\infty}$ norm, we first represent data in matrix form. Let $\mathbf{X}^v = [\mathbf{x}_1^v, \mathbf{x}_2^v, \cdots, \mathbf{x}_n^v]$ and $\boldsymbol{\alpha} = [\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \cdots, \boldsymbol{\alpha}_n]$. The dictionary learning problem is thus reformulated as

$$\min_{\mathbf{D}^v, \boldsymbol{\alpha}} H(\mathbf{D}^v, \boldsymbol{\alpha}) = \frac{1}{NV} \sum_{v=1}^{V} \|\mathbf{X}^v - \mathbf{D}^v \boldsymbol{\alpha}\|_{\text{Fro}}^2 + \lambda \|\boldsymbol{\alpha}\|_{1,\infty} \tag{5}$$

$$s.t. \|\mathbf{d}_j^v\| \leq 1, \ 1 \leq j \leq k$$

where $\|\cdot\|_{\text{Fro}}$ is the Frobenius norm. The $l_{1,\infty}$ norm is defined as $\|\boldsymbol{\alpha}\|_{1,\infty} = \sum_{i=1}^{k} \max_j |\alpha_i^j|$, where α_i^j is the (i,j)-th element in $\boldsymbol{\alpha}$.

The property of $l_{1,\infty}$ norm naturally leads to entire rows of $\boldsymbol{\alpha}$ to be zeroed out. Therefore, the locations of non-zero elements will cluster into groups instead of being randomly distributed. Such grouping effect is observed to bring significant performance boost compared with traditional sparse coding scheme.

Algorithm 1. Multi-view Shared Dictionary Learning

Input: $\mathcal{X} = \{\mathbf{X}^1, \cdots, \mathbf{X}^V\}$.
Output: $\mathcal{D} = \{\mathbf{D}^1, \cdots, \mathbf{D}^V\}$.
Initialize: $\boldsymbol{\alpha}_0 \in \mathbb{R}^{k \times N}$, $\mathbf{X}_{cat} = [\mathbf{X}^1; \cdots; \mathbf{X}^V]$.

Iterate for $t = 0, 1, 2, \ldots$ until convergence:

1. For $v \leftarrow 1$ to V do

$$\mathbf{D}_t^v = \arg\min_{\mathbf{D}^v} \|\mathbf{X}^v - \mathbf{D}^v \boldsymbol{\alpha}_t\|_{\text{Fro}}^2, \ s.t. \ \|\mathbf{d}_j^v\| \leq 1$$

   End for
2. Concatenate dictionaries vertically

$$(\mathbf{D}_{cat})_t = [\mathbf{D}_t^1; \cdots; \mathbf{D}_t^V]$$

3. Compute $\boldsymbol{\alpha}_{t+1}$ by using Algorithm 2

$$\boldsymbol{\alpha}_{t+1} = \arg\min_{\boldsymbol{\alpha}} \frac{1}{NV} \|\mathbf{X}_{cat} - (\mathbf{D}_{cat})_t \boldsymbol{\alpha}\|_{\text{Fro}}^2 + \lambda \|\boldsymbol{\alpha}\|_{1,\infty}$$

The optimization of (SDL-GS) is solved by alternating optimization. At every iteration, we 1) solve each view-specific dictionary $\mathbf{D}^v$ with fixed $\boldsymbol{\alpha}$ by using Block Coordinate Descent [13] and then 2) compute $\boldsymbol{\alpha}$ by using Nesterov's optimal gradient scheme as in Algorithm 2.

We adopt a window based stopping criterion: for a given window size h, at every iteration t, we calculate the following ratio

$$r = \frac{\max \mathcal{W}_t - \min \mathcal{W}_t}{\max \mathcal{W}_t}$$

where the set $\mathcal{W}_t = \{H_{t-h+1}, \cdots, H_t\}$ consists of history objective values in a window. If $r < \theta$, where θ is a predefined value, the algorithm stops iterating.

We illustrate the detailed procedures in Algorithm 1. SDL-SC as in Eq. 3 is solved similarly, but with l_1 norm over $\boldsymbol{\alpha}$, thus more sensitive to noise.

3.2 Nesterov's Optimal Gradient Scheme for $l_{1,\infty}$ Regularization

Now, we present an accelerated gradient descent methhod [20] to solve the convex $l_{1,\infty}$ regularization problem

$$\min_{\boldsymbol{\alpha}} F(\boldsymbol{\alpha}) = f(\boldsymbol{\alpha}) + \Psi(\boldsymbol{\alpha})$$

$$= \frac{1}{NV} \sum_{v=1}^{V} \|\mathbf{X}^v - \mathbf{D}^v \boldsymbol{\alpha}\|_{\text{Fro}}^2 + \lambda \|\boldsymbol{\alpha}\|_{1,\infty} \tag{6}$$

In order to apply Nesterov's optimal gradient scheme, we first need to define the proximal function and the generalized gradient mapping

$$Q_L(\boldsymbol{\alpha}, \boldsymbol{\alpha}_k) = f(\boldsymbol{\alpha}_k) + \langle \nabla f(\boldsymbol{\alpha}_k), \boldsymbol{\alpha} - \boldsymbol{\alpha}_k \rangle$$

$$+ \frac{L}{2} \|\boldsymbol{\alpha} - \boldsymbol{\alpha}_k\|_{\text{Fro}}^2 + \lambda \|\boldsymbol{\alpha}\|_{1,\infty} \tag{7}$$

$$q_L(\boldsymbol{\alpha}_k) = \arg\min_{\boldsymbol{\alpha}} Q_L(\boldsymbol{\alpha}, \boldsymbol{\alpha}_k) \tag{8}$$

where L is the a positive constant.

The proposed algorithm then alternately updates 1) a latent embedding matrix sequence $\{\boldsymbol{\alpha}_k\}$ by generalized gradient mapping and 2) an aggregation matrix sequence $\{\boldsymbol{\beta}_k\}$ by linear combination of $\{\boldsymbol{\alpha}_k\}$

$$\boldsymbol{\alpha}_{k+1} = q_{L_{k+1}}(\boldsymbol{\beta}_k) \tag{9}$$

$$\boldsymbol{\beta}_{k+1} = \boldsymbol{\alpha}_{k+1} + \frac{c_k - 1}{c_{k+1}} (\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}_k) \tag{10}$$

where $\{c_k\}$ is a scalar sequence and L_k is adjusted in every iteration.

The detailed description of Nesterov's optimal gradient is illustrated in Algorithm 2. The stop criterion is the same as used in Algorithm 1. Computing $q_L(\boldsymbol{\beta}_k)$ in Algorithm 2 can efficientlybe done by projection onto the l_∞ ball. [18].

4 Experiment

In this section, we demonstrate the effectiveness of the proposed SDL-GS algorithm on a VIPeR dataset[6], comparing with the state-of-the-art approaches. VIPeR dataset is most popular Multi-view camera tracking public available dataset. This dataset contains 632 unique pedestrians and a total of 1264 images with two views per pedestrian.

Algorithm 2. Nesterov's Optimal Gradient Scheme

Input: λ, L_0, η, $\mathbf{X}_{cat} \in \mathbb{R}^{D \times N}$, $\mathbf{D}_{cat} \in \mathbb{R}^{D \times k}$, where D is the concatenated feature dimension as in Algorithm 1.
Output: $\boldsymbol{\alpha} \in \mathbb{R}^{k \times N}$.
Initialize: $\boldsymbol{\alpha}_0$, $\boldsymbol{\beta}_0 = \boldsymbol{\alpha}_0$, $c_0 = 1$.

Iterate for $k = 0, 1, 2, \ldots$ until convergence of $\boldsymbol{\alpha}_k$:

1. Set $\bar{L} = L_k$
2. While $F\left(q_{\bar{L}}\left(\boldsymbol{\beta}_k\right)\right) > Q\left(q_{\bar{L}}\left(\boldsymbol{\beta}_k\right), \boldsymbol{\beta}_k\right)$

$$\bar{L} = \eta \bar{L}$$

   End while
3. Set $L_{k+1} = \bar{L}$ and update

$$\boldsymbol{\alpha}_{k+1} = q_{L_{k+1}}(\boldsymbol{\beta}_k)$$
$$c_{k+1} = \frac{1 + \sqrt{1 + 4c_k^2}}{2}$$
$$\boldsymbol{\beta}_{k+1} = \boldsymbol{\alpha}_{k+1} + \frac{c_k - 1}{c_{k+1}}\left(\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}_k\right)$$

Implementation Details and Evaluation. Experiments on this dataset are carried out in two aspects: 1) to compare SDL-GS with other factorization models, such as SVD and SDL-SC; 2) to compare our approach with other paradigms, such as AdaBoost and metric learning. All the algorithms are evaluated in terms of the Cumulative Matching Characteristic (CMC) curve, the same as in [6].

The images in the VIPeR dataset are 128 pixels high and 48 pixels wide. We extract color histograms from 8×8 pixels blocks. The blockes are densely collected from a regular grid with 4-pixel spacing in the vertical direction and 8-pixel spacing in the horizontal direction. The blocks are overlapped, with a step size of 4 pixels. We use RGB and HSV color spaces and extract 8-bin histograms in each channel. The all the histograms over the grid are concatenated to represent an image. The feature vector size is 2232 for both RGB and HSV. We also adopt other shape and texture feature descriptors, including dense SIFT descriptors [21] and dense Local binary pattern (LBP) descriptors[22]. The blocks used in these features is 8×8 pixels with 4 pixels spacing. We concatenate all the feature descriptors together and then apply Principal Component Analysis (PCA) to obtain 60-dimension low-dimension representation. The reduced features are normalized to unit vectors.

Since our view transformation model need to be trained on each combination of probe view and gallery view, we separate datasets into training set and testing set in each subsets. We learn each VTM respected to the different combinations of probe and gallery views, and calculate the cumulative recognition rates based on matching results from each view transformation model. After transform, we conduct the template

matching between transformed data and gallery data to obtain the matching scores. The people number distribution of different combinations of probe viewpoints and gallery viewpoints are not balanced. We directly employ template matching on those combinations that contains only 1, 3 or 4 pairs data (e.g. probe view is 90 degree while gallery view is 0 degree, probe 45 and gallery 45).Since there are some result variations in splitting training sets and testing sets, we report the average performanec of the approaches

Comparison within Factorization Models To validate the effectiveness of proposed approach, we compare with the other factorization approaches, SVD and SDL-SC, on each subsets of VIPeR dataset. In each subset, the probe data all come from the same viewpoints while gallery data are collected in the same way. We select the half of the data to constitute the training set while half of the data to constitute the testing set. Recognition rates at different retrieving ranks are presented in Table 1. And SDL-GS is superior than sparse coding but some times it is comparable with SVD.

Table 1. Recognition performance comparison among SDL-GS, SDL-SC and SVD in different subsets of the VIPeR dataset. The parameters of SDL-GS and SDL-SC are tuned to optimal performance. In the Viewpoint Pairs column, $0°$ to $90°$ refers to a subset of data where probe samples are in the viewpoint angle $0°$ and gallery samples are in the viewpoint angle $90°$. The other two notations follow the same convention. "–" denotes the "not applicable" because the subset does not have enough samples.

Viewpoint Pairs approach	Rank							
	5	10	15	20	25	30	35	40
$0°$ to $90°$ SDL-GS	**0.267**	**0.400**	**0.500**	**0.608**	**0.692**	**0.742**	**0.775**	**0.800**
SDL-SC	0.242	0.367	0.450	0.525	0.608	0.692	0.750	0.783
SVD	0.242	0.367	0.450	0.525	0.633	0.725	0.7667	0.7917
$0°$ to $180°$ SDL-GS	0.098	0.431	**0.627**	**0.706**	0.745	0.824	**0.863**	0.922
SDL-SC	0.098	0.431	0.549	0.667	0.745	0.824	0.843	0.922
SVD	0.098	0.412	0.588	0.667	**0.765**	0.804	0.843	**0.941**
$45°$ to $135°$ SDL-GS	0.056	**0.417**	**0.556**	0.611	0.694	**0.778**	**0.917**	–
SDL-SC	0.028	0.361	0.500	0.583	0.694	0.722	0.889	–
SVD	0.056	0.361	0.528	0.611	0.694	0.750	0.889	–

Comparison with Other Paradigms. To validate the effectiveness of proposed approach on overall dataset, we compare with the LMNN [7] and ELF200 [6], which are the state-of-the-art approaches in VIPeR dataset. We use the implementation of LMNN in [23]. We also implement the ELF200 reported in [6]. We conduct the proposed approach on each subset of VIPeR dataset according to the viewpoints. The λ is set as 0.02 while the dictionary size is set as 250.

Fig. 2 shows the CMC curves of competing approaches. SDL-GS is observed to be superior or comparable than the supervised approach of metric learning. We emphasize that our approach is unsupervised and can be easily combined with supervised approaches to further achieve performance.
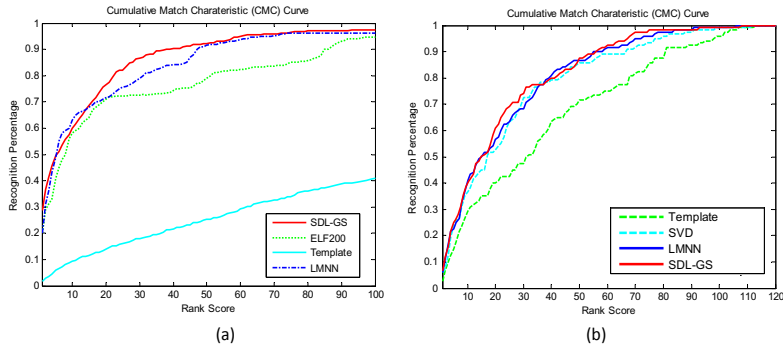
Fig. 2. Performance comparison among SDL-GS and other representative approaches. LMNN refers to the state-of-the-art approach reported in [7]. And ELF200 refers to [6]. Template is a baseline approach that directly compares distances between features of probe sample and gallery samples. (a) CMC curve comparison on all data; (b) CMC curve comparison on view pairs $0°$ to $90°$. Please refer to text (4) for detailed experimental setting.

5   Conclusion

In this work, we have presented a multi-view dictionary learning approach with shared latent embedding. Group sparsity constraints by $l_{1,\infty}$ regularization were adopted for more stable representation and superior performance. Our generative model provids a natural way to reconstruct data in unseen views. Such reconstruction ability is utilized in multi-view pedestrian and gait recognition. And various evaluation measures have demonstrated that SDL-GS outperforms other approaches in two mainstream datasets.

Currently, our model requires the knowledge of the viewpoint and that the training set should cover all viewpoints. In future work, we will investigate predicting viewpoint of the probe sample and training with incomplete views. We will also explore other extensions, such as by incorporating supervision information and learning Region of Interests (ROI).

References

1. Su, H., Sun, M., Fei-Fei, L., Savarese, S.: Learning a dense multi-view representation for detection, viewpoint classification and synthesis of object categories. In: ICCV, pp. 213–220 (2009)
2. Wu, B., Ram, N.: Cluster boosted tree classifier for multi-view, multi-pose object detection. In: ICCV, pp. 1–8 (2007)
3. Weinland, D., Boyer, E., Ronfard, R.: Action recognition from arbitrary views using 3d exemplars. In: ICCV, pp. 1–7 (2007)
4. Junejo, I.N., Dexter, E., Laptev, I., Pérez, P.: Cross-view action recognition from temporal self-similarities. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 1–8. Springer, Heidelberg (2008)
5. Prosser, B., Zheng, W.S., Gong, S., Xiang, T.: Person re-identification by support vector ranking. In: BMVC, pp. 21.1–21.11 (2010)

6. Gray, D., Tao, H.: Viewpoint Invariant Pedestrian Recognition with an Ensemble of Localized Features. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 262–275. Springer, Heidelberg (2008)

7. Dikmen, M., Akbas, E., Huang, T.S., Ahuja, N.: Pedestrian Recognition with a Learned Metric. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part IV. LNCS, vol. 6495, pp. 501–512. Springer, Heidelberg (2011)

8. Bashir, K., Xiang, T., Gong, S.: Cross-view gait recognition using correlation strength. In: BMVC, pp. 109.1–109.11 (2010)

9. Makihara, Y., Sagawa, R., Mukaigawa, Y., Echigo, T., Yagi, Y.: Gait Recognition using a View Transformation Model in the Frequency Domain. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3953, pp. 151–163. Springer, Heidelberg (2006)

10. Kusakunniran, W., Wu, Q., Li, H., Zhang, J.: Multiple views gait recognition using view transformation model based on optimized gait energy image. In: ICCV Workshop, pp. 1058–1064 (2009)

11. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. Trans. PAMI 31, 210–227 (2009)

12. Yang, M., Zhang, L.: Gabor Feature Based Sparse Representation for Face Recognition with Gabor Occlusion Dictionary. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6316, pp. 448–461. Springer, Heidelberg (2010)

13. Jenatton, R., Obozinski, G., Bach, F.: Structured sparse principal component analysis. In: AISTATS, pp. 366–373 (2010)

14. Efron, B., Hastie, T., Johnstone, I., Tibshirani, R.: Least angle regression. Ann. Stat. 32 (2004)

15. Lee, H., Battle, A., Raina, R., Ng, A.Y.: Efficient sparse coding algorithms. In: NIPS, pp. 801–808 (2007)

16. Huang, J., Huang, X., Metaxas, D.: Learning with dynamic group sparsity. In: ICCV, pp. 64–71 (2009)

17. Bengio, S., Pereira, F., Singer, Y., Strelow, D.: Group sparse coding. In: NIPS, pp. 82–89 (2009)

18. Chen, X., Pan, W., Kwok, J.T., Carbonell, J.G.: Accelerated gradient method for multi-task sparse learning problem. In: ICDM, pp. 746–751 (2009)

19. Jia, Y., Salzmann, M., Darrell, T.: Factorized latent spaces with structured sparsity. In: NIPS, pp. 982–990 (2010)

20. Nesterov, Y.: Gradient methods for minimizing composite objective function. Technical report, Euro. (2007)

21. Vedaldi, A., Fulkerson, B.: Vlfeat – an open and portable library of computer vision algorithms. In: ACM Multimedia, pp. 1469–1472 (2010)

22. Ojala, T., Pietik1in, M., Harwood, D.: Performance evaluation of texture measures with classification based on kullback discrimination of distributions. In: ICPR, pp. 582–585 (2004)

23. Weinberger, K., Blitzer, J., Saul, L.: Distance metric learning for large margin nearest neighbor classification. In: NIPS, pp. 1473–1480 (2006)

A Feature Selection Approach for Emulating the Structure of Mental Representations

Marko Tscherepanow[1,4], Marco Kortkamp[1], Sina Kühnel[2,4],
Jonathan Helbach[1], Christoph Schütz[3,4], and Thomas Schack[3,4]

[1] Applied Informatics, Faculty of Technology
[2] Physiological Psychology, Faculty of Psychology and Sport Sciences
[3] Neurocognition and Action, Faculty of Psychology and Sport Sciences
[4] CITEC, Cognitive Interaction Technology, Center of Excellence
Bielefeld University, Universitätsstraße 25, 33615 Bielefeld, Germany
{marko,mkortkam}@techfak.uni-bielefeld.de,
skuehnel@cit-ec.uni-bielefeld.de

Abstract. In order to develop artificial agents operating in complex ever-changing environments, advanced technical memory systems are required. At this juncture, two central questions are which information needs to be stored and how it is represented. On the other hand, cognitive psychology provides methods to measure the structure of mental representations in humans. But the nature and the characteristics of the underlying representations are largely unknown. We propose to use feature selection methods to determine adequate technical features for approximating the structure of mental representations found in humans. Although this approach does not allow for drawing conclusions transferable to humans, it constitutes an excellent basis for creating technical equivalents of mental representations.

Keywords: Feature selection, Mental representations, Memory.

1 Introduction

One of the biggest challenges today is the endeavour to copy or emulate memory as it is found in humans and animals. In principle, memory constitutes the basis for any kind of learning to be performed. Therefore, a multitude of approaches related to the topic of memory in artificial systems have been proposed. They adopt single properties of natural memory, in particular, its structure [1], its processes [2], or mental representations [3].

A crucial problem with developing artificial agents using memory systems is the formation of appropriate technical representations of perceptual data. Similar to natural agents possessing cognitive capabilities, technical memory systems have to obey the principle of cognitive economy [4]; i.e., the amount of data needs to be diminished before it is stored. Otherwise, the deluge of incoming sensory information would quickly consume the entire memory. Nevertheless,

the formed representations need to contain the relevant information. Two important methods for achieving this goal are the formation of categories [4] and dimensionality reduction [5].

The goal of our work consists of the emulation of the structure of human mental representations by means of features that can be computed from visual stimuli (images). In order to comply with the principle of cognitive economy, the resulting feature sets should be as small as possible. Therefore, several feature selection methods are compared. As the selected features contain the information to replicate the results obtained from humans, we assume that they are good candidates for representing the corresponding images in artificial systems.

In Section 2, we introduce different methods for analysing mental representations in humans. Afterwards, popular feature reduction methods are discussed in Section 3. Our complete approach is described in Section 4 and evaluated in Section 5. Finally, Section 6 summarises the most important outcomes.

2   Psychological Background

One way to obtain knowledge about human mental representations consists of conducting experiments in which subjects assign labels to perceived stimuli (e.g., [6]). From these, conclusions about the internal concepts and features used for classification can be drawn. But degrees of class membership are usually not reflected. In [7], a method explicitly avoiding semantic groups was applied: The subjects successively split presented images into two groups. Here, images of one group should share a common global aspect, structure, or certain elements. Afterwards the subjects were asked to verbally describe the splitting criteria used. Hence, the features utilised for splitting were associated with a label, e.g., naturalness, which itself represents a concept.

Structural Dimensional Analysis (SDA) [8] constitutes an alternative approach to the analysis of mental representations. In contrast to the methods introduced above, it does not require labels provided by subjects. In cognitive psychology, SDA is a well-established method for psychometrically investigating the representational structure of concepts in long-term memory. The concepts under analysis are verbally defined by the experimenter, e.g., 'wood', 'brush', and 'hat' [8]. This original SDA method was extended to the analysis of the representational structure of motor skills, which is called Structural Dimensional Analysis-Motoric (SDA-M) [9,10]. The extension from verbally defined concepts to movements was achieved by introducing so-called basic action concepts (BACs), which represent components of complex movements that are characterised by perceivable features. In this context, SDA was shown to work with visual stimuli as an alternative to verbal descriptions.

3   Relevant Feature Reduction Methods

For numerous machine learning techniques, a feature reduction step is required in order to avoid problems arising from the *curse of dimensionality* [11]. Feature

reduction methods aim to decrease the dimensionality of the feature space while minimising the information loss. This is achieved by removing irrelevant and redundant information. In general, feature reduction can be divided into two principal methods: feature extraction (e.g., principal component analysis [12] and independent component analysis [12]) and feature selection (e.g., minimum-redundancy-maximum-relevance [13] and genetic algorithms [14]). Feature extraction computes a functional mapping for which the underlying meaning of the features is lost. In contrast, feature selection chooses salient features from the original feature set and thereby preserves the features' semantics. Moreover, unselected features do not need to be computed. For these reasons, we decided to focus on feature selection rather than feature extraction approaches.

Feature selection methods can be divided into filters and wrappers. While filters operate independently of the utilised machine learning technique and optimise pre-selected criteria [11,13], wrappers optimise the actual learning results provided by an induction algorithm [14]. As a consequence, wrappers often lead to better results while filters are less computationally expensive [11]. Additionally, feature selection approaches can be distinguished depending on the way they determine sets of relevant features. Some methods first measure the quality of individual features and rank them [15]. Then, the top-ranked features are selected. Since the actual number of required features is hard to choose, other approaches directly select feature subsets [11,13]. These feature subset selection methods usually provide better results than ranking-based techniques, since they account for redundancies and complex interdependencies of the considered features [13].

Structural Dimensional Analysis itself can identify common features in different representational units (items) [8,9,10]. This is achieved by means of factor analysis [12], which is frequently applied in psychology. It provides meaningful factors explaining observed results and is closely related to the feature extraction methods mentioned above. However, the factors themselves cannot be computed from the stimuli. They rather are unobservable variables describing the experimental results obtained from human subjects.

4   Our Approach

An overview of our approach is given in Fig. 1. First, SDA is performed in order to measure the representational distances of a set of images in different human subjects (cf. Section 4.1). Then, feature-based representations of these images are determined. In principle, any kind of real-valued features which can be computed from individual images could be applied here. From this initial set, subsets of features are selected. An initial set or a selected subset are considered valid, if they enable an accurate mapping to the measured representational distances; i.e., if such a mapping exists, we assume that the respective features contain all information required for the reproduction of the representational distances found in humans. But the mapping itself may be very complex and non-linear.

In order to test for the existence of a mapping from the feature-based representations of the images to the measured representational distances, we attempt
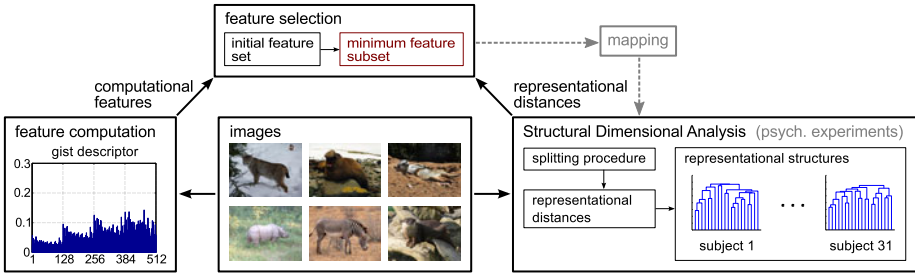
Fig. 1. Principal approach. A feature selection method chooses minimum sets of computable features in order to determine an efficient technical representation of natural images. The representational distances determined by SDA serve as ground truth, which is to be approximated using the respective feature subsets.[1]

to compute an adequate regression model. Provided that such a model has been found, it is concluded that the applied feature set suffices to represent the images under consideration. In contrast to traditional machine learning approaches, the application of distinct test and training datasets is neither possible nor necessary for the training of the regression models. Firstly, such training sets would not be representative for the complete input distribution, as the underlying human information processing is too complex and results in unpredictable representational distances between untrained stimuli. Secondly, we aim at explaining observed data, similar to SDA, and do not require good generalisation properties of the regression models. But unlike the regression models, the determined feature subsets are validated.

As the maximum number of images and, therefore, the amount of available samples is very limited due to the algorithmic properties of SDA (see Section 4.1), we decided to apply Support Vector Regression[2] (SVR) [16]. Regarding the task of feature selection, several methods are compared (see Section 4.2).

4.1   Generating Ground Truth Data – SDA

In a first step, SDA seeks to gain information about the distance between representational units corresponding to a set of n_s selected stimuli. Since the structure of mental representations can only be explicated by subjects to a limited extent, this is achieved by a special splitting technique: one stimulus is chosen as an anchor and the remaining stimuli are compared to it (in random order) and manually classified as 'similar' or 'dissimilar'. This is repeated for the resulting subsets until they become too small to be split or the subject decides that further splitting is not reasonable. Thus, a decision tree is constructed.

---

[1] Lynx, sea-elephant, meerkat, and otter: CC-by-SA 3.0 Unported; rhinoceros and zebra: CC-by-A 2.0 Generic.

[2] We used the ν-SVR implementation of LIBSVM, version 3.0.

The splitting procedure is repeated in such a way that each stimulus serves as an anchor. Therefore, the number of constructed decision trees equals the number of stimuli.

In order to obtain a distance measure, the algebraic sums along all branches are computed for each decision tree. Here, stimuli classified as 'dissimilar' obtain a negative sign and elements classified as 'similar' a positive sign. From the resulting values, a matrix is constructed, with its elements s_{ik} denoting the sum for stimulus k with respect to anchor i. These sums are z-transformed:

$$z_{ik} = \frac{s_{ik} - \mu_i}{\sigma_i} \quad , \text{with} \quad \mu_i = \frac{1}{n_s} \sum_{k=1}^{n_s} s_{ik} \quad \text{and} \quad \sigma_i = \sqrt{\frac{1}{n_s} \sum_{k=1}^{n_s} (s_{ik} - \mu_i)^2}. \quad (1)$$

Then, a correlation matrix is computed. The individual correlation r_{ij} of two stimuli i and j is further transformed into the Euclidean distance measure d_{ij}:

$$d_{ij} = \sqrt{2n_s}\sqrt{1 - r_{ij}} \quad , \text{with} \quad r_{ij} = \frac{1}{n_s} \sum_{k=1}^{n_s} z_{ik} z_{jk}. \quad (2)$$

The computed distances d_{ij} are subjected to a hierarchical cluster analysis which reveals the representational structure of the stimuli and constitutes the second step of SDA. As the mental representations differ between the individuals of a population, the measured structures exhibit differences as well. The third step comprises a cluster-dependent factor analysis revealing underlying dimensions in the structured set of representations and the final step consists in testing for invariance within and between groups of subjects.

Our work focusses on the first step, as the distance values provided therein completely define the representational structure revealed by cluster analysis. Since d_{ii} always equals 0 independent of the representational structure and the underlying representations, we decided to omit these values. Furthermore, the number n_d of available distance values is reduced due to symmetry ($d_{ij} = d_{ji}$). It amounts to $\frac{1}{2} n_s(n_s - 1)$. Due to the high number of comparisons which have to be performed by the subjects (up to $O(n_s^3)$), the number of obtainable distance values is very limited. In particular, n_s should not be chosen higher than 20. Otherwise, the decisions made regarding the similarity of stimuli may become inconsistent.

4.2 Feature Selection

In order to select adequate features, it must be considered that each subject has individual mental representations of the images and, therefore, the mappings from the stimuli to their representations and the resulting representational structures may vary considerably. Nevertheless, it would be beneficial if the data from different subjects could contribute to a common feature subset, as the amount of available data is considerably increased this way. Furthermore, we assume that the principal way of information processing does not differ considerably between

different healthy human subjects. Thus, our approach aims at aligning the feature subsets found for all subjects, in addition to pursuing the traditional goal of minimising the number of selected features.

The nature of the task at hand implies the usage of a wrapper approach, as the quality criterion consists of the accurate approximation of the representational distances d_{ij} provided by SDA. Hence, we decided to apply a genetic algorithm [17], due to the flexibility of this method. The developed algorithm is specifically tailored to the problem at hand. Nevertheless, it would be advantageous if standard methods could be applied as well. Therefore, we analysed two further feature selection approaches, namely Correlation-based Feature Selection (CFS) [11] and ReliefF [15].[3] As these methods are filters, we expected them to be less computationally expensive than the genetic algorithm. But they are not able to process the subjects individually while simultaneously aligning their results. Hence, we applied these methods to the collective data of all considered subjects, in order to find a single feature subset.

Genetic Algorithm. In our genetic algorithm (GA), a candidate solution, also called an individual, constitutes a combination of a feature subset and an associated regression model approximating the representational distances of a specific subject. As a result, the genome of each individual comprises two components: (i) the feature genome $\underline{g}^f$ defining the selected features (and the dimensionality of the feature space) and (ii) the parameter genome $\underline{g}^p$ defining the parameters for the SVR. Here, three possible kernels – linear, radial basis function (RBF), and sigmoid – are considered depending on the parameter $type$.

While feature subsets are defined by binary genes denoting whether a specific feature is selected or not, the SVR parameters are encoded as numerical values from the interval [0,1]. For the regularisation constant C and the kernel parameters γ, κ, and ϑ, these numerical genes are mapped to the interval [0.00001, 10000]. The feature genome is adapted by bit mutation [17] with the probability p_m and uniform cross-over [17]. For the parameter genome, a mutation operator for real-valued genes[4] [18] and arithmetic cross-over [18] are utilised. p_c denotes the cross-over probability for both operators. In the initial generation, features are randomly selected with the mutation probability p_m.

In order to align the feature sets selected for different subjects, each feature i is assessed by a weight

$$w(i) = \frac{\sum_{A \in \mathcal{E}} g_A^f(i)}{\sum_{j=1}^{n_f} \sum_{A \in \mathcal{E}} g_A^f(j)} \tag{3}$$

reflecting the frequency of its occurrence in the set $\mathcal{E}$, which summarises the elite individuals of the current generation for all subjects. Here, n_f denotes the number of features.

The three goals explained above are reflected by the fitness function which is used for evaluating the performance of each individual A:

---

[3] For CFS and ReliefF, the implementations of WEKA, version 3.6.3, were used.
[4] Changes are sampled from the Gaussian $\mathcal{N}(0, 0.025^2)$.

$$F(A) = 1 - (1 - c_f - c_h) \underbrace{E(A)}_{(i)} - c_f \underbrace{\frac{1}{n_f} \sum_{i=1}^{n_f} g_A^f(i)}_{(ii)} - c_h \underbrace{\left(1 - \sum_{i=1}^{n_f} g_A^f(i) w(i)\right)}_{(iii)}. \quad (4)$$

Component (i) minimises the regression error $E(A)$, component (ii) minimises the size of the chosen feature subset, and component (iii) assures the alignment of selected features across all subjects.

The constants should be chosen as follows: $1 \gg c_f \gg c_h$. By this, the regression error obtains the highest priority, followed by the feature set size and the alignment of feature sets between different subjects. As the influence of the components (ii) and (iii) is very small compared to the regression error $E(A)$, we applied rank-based selection [18].

The final feature subset consists of those features which were applied by all elite individuals of the final generation.

5   Results

The suitability of the considered feature selection methods was analysed using 20 images (400×300 pixels) showing different, complete and centred animals in their natural environment (cf. Fig. 1). The representational distances between these images were measured for $s=31$ subjects (16 male, 15 female; age: 21–46) resulting in $n_d=190$ samples per subject and $5,890$ samples in total. In order to alleviate the evaluation, the distances of each subject were normalised to the interval $[0,1]$. As an example for feature-based image representations, we employed the well-established gist descriptor ($n_f=512$) [7]. For the evaluation, ten different splittings of the set of subjects into subsets of 20 training subjects and 11 test subjects each were randomly created. The data of the respective training subjects is applied so as to select salient features. These suitability of these features is tested with respect to the test subjects.

For each training subject, the genetic algorithm optimised 100 individuals, the 10 fittest of which were regarded as elite individuals.[5] Figure 2 depicts the development of the mean regression error $\mu(\overline{E})$ for the respective training subjects and the mean number of selected features $\mu(\overline{n}_{sf})$ during the optimisation. In addition, the mean normalised histogram intersection $\mu(\overline{H})$ [19] is plotted.[6] It measures the similarity between the feature usage histograms of the elite individuals assigned to different subjects. Figure 2 demonstrates that the genetic algorithm achieves the goals stated in Section 4.2.

---

[5] The remaining parameters were set to the following values: $p_c=0.25$, $c_f=0.01$, and $c_h=0.0005$.

[6] Over-lined symbols denote average values over all elite individuals and/or training subjects, while μ and σ denote the mean and the standard deviation over the different splittings, respectively.
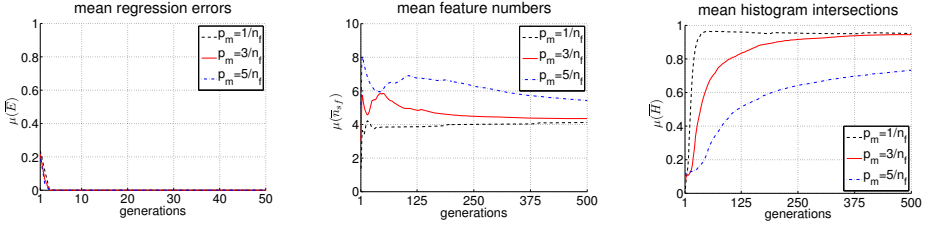
Fig. 2. Results of the evolutionary optimisation performed by the genetic algorithm. The regression error decreases rapidly during the first generations (left). The feature number (centre) and the alignment of the features subsets of different subjects (right) require more generations to converge. Here, larger values of the mutation probability p_m retard the optimisation process.
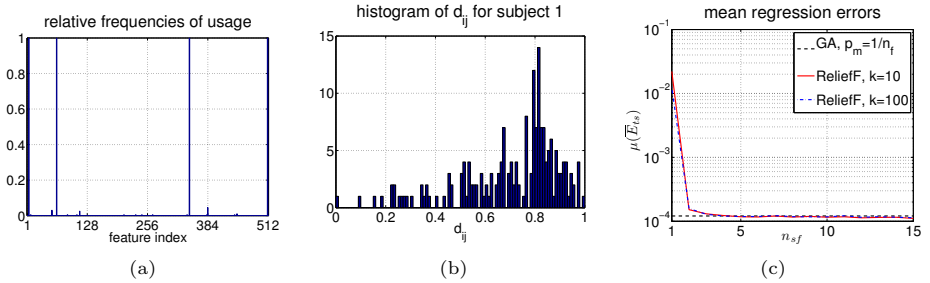
Fig. 3. Relative frequencies of feature usage by the elite individuals after a single run of the genetic algorithm (a), distribution of the normalised representational distances d_{ij} for a single subject (b), and mean regression errors of ReliefF for the test subjects depending on the number of selected features n_{sf} (c).

In order to determine the final feature subset, the relative frequencies of the usage of features by the elite individuals of the final generation are analysed. An exemplary result is shown in Fig. 3(a). Those four features, which were used by all elite individuals, constitute the resulting feature subset.[7]

The validity of the chosen feature subsets was tested using the data of the respective test subjects (see Table 1). The parameters ν, C, and γ for the SVR (RBF kernel) were determined by grid search (11 values per parameter) individually minimising the regression error E_{ts} for each test subject. The genetic algorithm was compared to CFS and ReliefF. In case, the computation of the regression models did not terminate using the default criterion ($\epsilon=0.001$), the respective splitting was omitted.[8] The results for CFS using the default parameters and different search directions are given in Table 1, as well. Here, it must be

---

[7] Due to the random nature of the genetic algorithm and redundancies in the initial feature set, the actually selected features varied across different trials. But their number was approximately constant.

[8] CFS, forward: splitting 4; ReliefF, $k=100$: splittings 6 and 8.

Table 1. Means μ and standard deviations σ of the regression errors E_{ts} for the test subjects and the corresponding sizes n_{sf} of the chosen feature subsets.

feature selection approach	$\mu(\overline{E}_{ts})$	$\sigma(\overline{E}_{ts})$	$\mu(\overline{n}_{sf})$	$\sigma(\overline{n}_{sf})$
genetic algorithm, $p_m=1/n_f$	$1.21 \cdot 10^{-4}$	$6.07 \cdot 10^{-6}$	3.9	0.7
genetic algorithm, $p_m=3/n_f$	$1.25 \cdot 10^{-4}$	$8.92 \cdot 10^{-6}$	4.1	0.7
genetic algorithm, $p_m=5/n_f$	$1.23 \cdot 10^{-4}$	$9.57 \cdot 10^{-6}$	3.9	0.7
CFS, forward	$1.17 \cdot 10^{-4}$	$6.52 \cdot 10^{-6}$	11.0	2.67
CFS, backward	$1.17 \cdot 10^{-4}$	$5.34 \cdot 10^{-6}$	13.5	2.06
CFS, bi-directional	$1.15 \cdot 10^{-4}$	$5.71 \cdot 10^{-6}$	10.2	1.99

considered that the majority of the distances d_{ij} is centred around a single peak (see Fig. 3(b)). Therefore, very small errors are required in order to preserve the representational structure.

Both the genetic algorithm and CFS enable the approximation of the representational distances with high accuracy. But the feature subsets determined by CFS are larger. This is likely to be a result of the collective processing for all training subjects.

In contrast to our approach and CFS, ReliefF does not directly select feature subsets but provides quality assessments and a ranking. Figure 3(c) depicts the mean regression error depending on the number of selected features using two different neighbourhood sizes k. If the 4 top-ranked features are selected, the regression errors are comparable to the genetic algorithm and CFS. A further increase of the feature set size does not lead to significant improvements, although ReliefF collectively processed the data of all training subjects like CFS.

6   Conclusion

We compared several feature selection methods regarding their ability to select subsets of computable features enabling the emulation of the structure of mental representations found in humans. Standard feature selection methods, in particular CFS and ReliefF, achieved results comparable to a genetic algorithm that was specifically tailored to this problem. Using such methods, the results of SDA can be explained in terms of small sets of salient features which are directly computable from the stimuli. In the future, the resulting feature sets could be exploited to learn human-like representational structures in technical agents. For example, adequate feature subsets could be determined off-line. As they preserve the relevant information, their usage instead of the original stimuli would not reduce the potential learning capabilities of the agent during interaction with its environment. However, the amount of data to be stored would be considerably reduced.

References

1. Kawamura, K., Gordon, S.M., Ratanaswasd, P., Erdemir, E., Hall, J.F.: Implementation of cognitive control for a humanoid robot. International Journal of Humanoid Robotics 5(4), 547–586 (2008)
2. Ben Amor, H., Ikemoto, S., Minato, T., Jung, B., Ishiguro, H.: A Neural Framework for Robot Motor Learning Based on Memory Consolidation. In: Beliczynski, B., Dzielinski, A., Iwanowski, M., Ribeiro, B. (eds.) ICANNGA 2007. LNCS, vol. 4432, pp. 641–648. Springer, Heidelberg (2007)
3. Chartier, S., Giguère, G., Langlois, D.: A new bidirectional heteroassociative memory encompassing correlational, competitive and topological properties. Neural Networks 22(5-6), 568–578 (2009)
4. Goldstone, R.L., Kersten, A.: Concepts and categorization. In: Healy, A.F., Proctor, R.W. (eds.) Handbook of Psychology, vol. 4: Experimental Psychology, pp. 599–621. John Wiley & Sons, Hoboken (2003)
5. Edelman, S., Intrator, N.: Learning as extraction of low-dimensional representations. In: Goldstone, R.L., Schyns, P.G., Medin, D.L. (eds.) Perceptual Learning, pp. 353–380. Academic Press, San Diego (1997)
6. Johansen, M.K., Kruschke, J.K.: Category representation for classification and feature inference. Journal of Experimental Psychology 31(6), 1433–1458 (2005)
7. Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. International Journal of Computer Vision 42(3), 145–175 (2001)
8. Lander, H.J., Lange, K.: Untersuchungen zur Struktur- und Dimensionsanalyse begrifflich repräsentierten Wissens. Zeitschrift für Psychologie 204, 55–74 (1996)
9. Bläsing, B., Tenenbaum, G., Schack, T.: The cognitive structure of movements in classical dance. Psychology of Sport and Exercise 10, 350–360 (2009)
10. Schack, T., Mechsner, F.: Representation of motor skills in human long-term memory. Neuroscience Letters 391, 77–81 (2006)
11. Hall, M.A.: Correlation-based Feature Selection for Machine Learning. Ph.D. thesis, Department of Computer Science, The University of Waikato, Hamilton, New Zealand (1999)
12. Hyvärinen, A., Karhunen, J., Oja, E.: Independent Component Analysis. John Wiley & Sons, New York (2001)
13. Peng, H., Long, F., Ding, C.: Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(8), 1226–1238 (2005)
14. Yu, L., Chen, H., Wang, S., Lai, K.K.: Evolving least squares support vector machines for stock market trend mining. IEEE Transactions on Evolutionary Computation 13(1), 87–102 (2009)
15. Kononenko, I., Šikonja, M.R.: Non-myopic feature quality evaluation with (R)ReliefF. In: Liu, H., Motoda, H. (eds.) Computational Methods of Feature Selection, pp. 169–191. Chapman and Hall, Boca Raton (2008)
16. Schölkopf, B., Smola, A.J., Williamson, R.C., Bartlett, P.L.: New support vector algorithms. Neural Computation 12, 1207–1245 (2000)
17. Fogel, D.B.: Evolutionary Computation, 3rd edn. IEEE Press, Piscataway (2006)
18. Engelbrecht, A.P.: Computational Intelligence, 2nd edn. John Wiley & Sons, Hoboken (2007)
19. Swain, M.J., Ballard, D.H.: Color indexing. International Journal of Computer Vision 7(1), 11–32 (1991)

Super Resolution of Text Image by Pruning Outlier

Ziye Yan, Yao Lu, and JianWu Li

Beijing Laboratory of Intelligent Information Technology,
School of Computer Science & Technology,
Beijing Institute of Technology, Beijing, China
{yanziye,vis_yl,ljw}@bit.edu.cn

Abstract. We propose a learning based super resolution algorithm for single frame text image. The distance based candidate of example can't avoid the outliers and the super resolution result will be disturbed by the irrelevant outliers. In this work, the unique constraints of the text image are used to reject the outliers in the learning based SR algorithm. The final image is obtained by the Markov random field network with k nearest neighbor candidates from an image database that contains pairs of corresponding low resolution and high resolution text image patches. We demonstrate our algorithm on simulated and real scanned documents with promising results.

Keywords: Super resolution, Text image, Markov random field, Outlier.

1 Introduction

The single frame text image super resolution (SR) is to estimate missed higher resolution details from only one observed low resolution (LR) text image. In general, the super resolution reconstruction method requires multiple LR images captured from the same scene with sub-pixel shift [1]. When the input images are insufficient or the magnification factor is increased, the reconstruction will fail. Baker analyzed the constraints of the SR and proposed the learning based super resolution [2]. In the learning based SR method, the prior knowledge extracted from the train dataset is used as the additional constraints, and the input image number can reduce to only one. The basic idea of the learning based SR method is that the image is reconstructed from the k nearest neighborhood (kNN) found candidates by maximum posterior (MAP) or minimum mean square error (MMSE) estimation [2-6]. The neighborhood constraint is introduced into the Markov random field to reconstruct the image with local compatibility [2, 3]. Performance of their works lays in the large training database, allowing rich prior probabilities. As the kNN examples include many outliers, Elad proposed a global pruning outlier method within the MAP scope [4]. The global MAP penalty function only prunes one outlier in each iterative, and the edge is blurred for the reconstruction without any local constraints. In [5] the example database is separated into several classes, and the LR input patch has been classified into one class before reconstruction. The predictor will be the bottleneck of this method. In [6] the special prior such as the edge also is used for improve the match processing.

Since the non-negligible null-space of the degradation operator, the outliers cannot be avoided in the MSE search [4]. The additional constraints are required for the pruning, which is hardly obtained from the generic image. In contrast to the task of generic image SR, there are stronger constraints in the text images which provide the chance to obtain vivid character output. In this paper, we present a new neighbor constraint model for text image in Bayesian framework via the MRF. The unique constraints of text image are taken account into pruning the outlier of the selected candidates before reconstruction. This pruning is based on the local feature contrasting with [4], and all of the example in database will be looked thought to overcome the drawback of predictor in [5]. After the outliers in the kNN examples have been rejected, the computational complexity of the MRF decreases. Moreover, the huge duplicate database is replaced by a special small reduced example database.

The paper is organized as follows. In Section (2) the framework of the learning based SR is introduced and the constraints of text super resolution is analyzed; the outliers in the learning based SR is discussed in this section. In Section (3), the proposed method is given. Experiments are then provided and discussed in Section (4) and the paper concludes in Section (5).

2   Learning Based Super Resolution and the Super Resolution Constraints

2.1   The Framework of Learning Based Super Resolution

The learning based SR in Bayesian framework is proposed as finding the MAP or MMSE from the following formulation [2].

$$P(S \mid L) = \frac{P(L \mid S) \cdot P(S)}{P(L)} \tag{1}$$

where $P(S)$ is the prior knowledge. $P(L|S)$ is the estimation of low-resolution image from HR in image formation model. $P(S|L)$ is the posterior probability density function.

The neighbor-constraint knowledge comes from LR and HR image patch pairs based on the Markov network. Markov network used in this paper models the statistical relationships between LR and their corresponding HR patches and the statistical relationships between HR patches and their overlapping neighbor ones. The structure of the Markov network and the neighborhood considered for calculations are shown in Fig.1. The joint compatibility over observed node y (input LR patch) and selected example x (LR and corresponding HR patch pairs) is as below [3]

$$P(x \mid y) = \frac{1}{Z} \prod_{neighboring(k,j)} \Psi(x_k, x_j) \prod_k \Phi(x_k, y_k) . \tag{2}$$

where the compatibility function $\Phi(x_k, y_k)$ is the exponential of the error between the observed node (input LR patch) and the LR patch of the selected example pairs, and $\Psi(x_k, x_j)$ is the exponential of matching pixels ratio of neighboring patches overlap in

the HR patch. It should be noted that the x_k in the Φ and Ψ will use the LR and HR patch of the selected example pair separately, which is the tie of LR and HR image. The MAP and MMSE for x_j at node j are

$$\hat{x}_j MAP = \arg \max_{x_j} \max_{allx_i, i \neq j} P(x, y) \; . \tag{3}$$

$$\hat{x}_j MMSE = \int_{x_j} x_j dx_j \int_{allx_i, i \neq j} P(x, y) dx \; . \tag{4}$$

The optimal solution of this MRF can be obtained using the belief propagation algorithm [3].
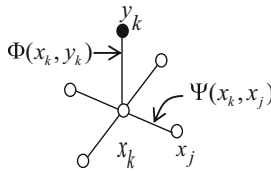
Fig. 1. The MRF structure

2.2   The Degrading Constraint

In generally, the low resolution input images are considered as degraded result of a high resolution image. The mathematical expression is as below [1]

$$l = degrade(h) = T(h * PSF) \downarrow s + \eta \; . \tag{5}$$

where l is LR observed images, h is HR image, T is geometric transformation, η is additive noise, $\downarrow s$ is down sampling, PSF is Point Spread Function (PSF).

For the multiple images SR reconstruction, the constraint includes the PSF and the registration of each LR input [2]. The patch pairs used in the learning method may be regarded as combination constraint of the PSF and registration although only one input LR image.

2.3   The Unique Constraint of Text Image

Beyond the general image, the text image has the unique constraints as the positive information which is contained in the text. Those constraints are list in follow:

The low resolution: as the optical character recognition (OCR) has the ability to recognize the text from the high quality text image, the super resolution is restricted to the low resolution in which the traditional or the commercial OCR has failed [2].

The fixed content: the content of text image is the character which has fixed shape. Although the size and the fold are variable, the patterns of character are finite. So less inaccuracy of the result will be tolerated than the general image. Moreover, location relationship of the stroke or the patch in the image is fixed or invariable.

The few gray level: unlike the nature image, the text image has bimodal distributions in the gray level [7]. The high quality text image may be clearly expressed in binary; while in the low resolution, the text also can be expressed in a few gray levels.

The sharp edge: in the high quality text image, the edge of stoke is sharp, and the profile of stroke is smooth and continuous.

2.4   The Outlier in Learning Based SR

The outlier in the learning based SR is been defined in follow:

Definition 1. let $P = (P_l, P_h)$ is an patch pair in the example database, P_l and P_h is the LR and HR patch, the lp is one patch extracted from LR observed image l, the l is obtained from HR image h by (1), hp is the corresponding HR patch. If P is one of the selected candidates based on $\Phi(P_l, lp)$ and $d(P_h, hp) > t$, then P is outlier. Where d is the distance, t is the threshold. In this work, we use the $L2$ norm as the distance.

While the outlier problem has been discussed in [4], the authors indicated that the outliers are unavoidable even if all examples were close in the LR. In this work the outliers in different location of the image is concerned. As shown in Fig. 2, the percent of outlier will increase in the edge region, and in the inner of one character the outlier is decreased. In additional, the outliers will be uniform in the region out of the character or in the background because the database contains abundant blank patches.

a)The LR image 100dpi

b) The true HR image 200dpi   c) The percent of outlier in each position
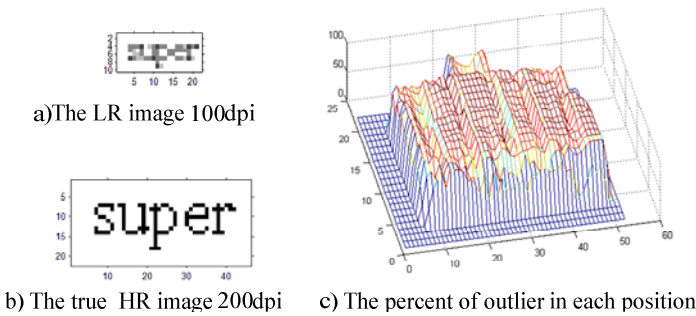
Fig. 2. The percent of outlier of text image, the position in the x-y plane is the location of the patches in the image, and the z axis gives the percent of outlier in the top 50 candidates
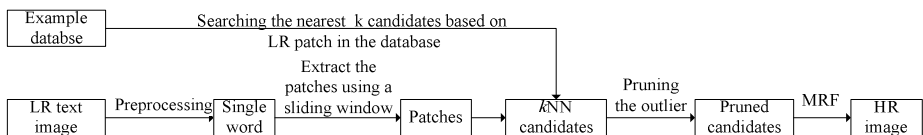
Fig. 3. The flow chart of the proposed method

3   The Proposed Super Resolution Method

As the outlier is inevitable in the low resolution match step, we propose a method to prune the outlier using the unique constraint of text image. The framework of this method is shown in Fig.3.

3.1   Preprocess of Low Quality Image

This step includes the intensity adjustment and segmentation. The LR image usually is described in gray image and the intensity will vary by the various capture condition. These factors will increase computational complexity and produce imprecise matching particularly with the reduced duplicate or insufficient database. Because the text image may be described with less number of gray levels, four gray levels is used which in enough to describe the bimodal distributions in our experiments. Although the LR image is hardly segmented into the letter, it is easily extracted every word through the projection processing.

3.2   Pruning the Outlier

From the (2), it is shown that the probability of each node will depend on all the k states (kNN candidates) of this node. And the computational complexity of the Markov network will increase with the $O(k^2)$. As discussed above, there are many outliers in the k found examples. Thus the correct candidate will be flooded into the outlier in particular the dataset has been reduced the duplicate.

In Freeman's works, the database contained many duplicate candidates [3]. In [4] the outlier was pruned in global under the likelihood penalty function and the edge is blurred for the reconstruction without any local constraints. In this work, the unique constraints of text image are used to pruning the outlier before constructing the network.

a) The background pruning. Although the LR is low quality, the background is easy detected. We use following rules to detect the background:

● *The segmented image is projected in horizontal and vertical direction. The nodes in the region out of the center rectangle are background.*

● *If all pixels of the node are white, it is an background candidate;*

● *If at least two 4-connected neighborhoods of the candidate node are background, the node is regarded as background.*

The result of background pruning is shown in Fig. 5(a). The input LR image is shown in Fig.2 (a). The region out of the contour of the word has been considered as the background.

b) The pruning with structural feature. The text image has constant structural in every character. So the outlier can be found out using those constraints.

● *The pruning with location feature.* Since the stroke of the LR image is hardly extracted, the localization feature is used to prune the outliers. The localization feature concerns about the distributing of the strokes. We use the center line as the reference line, and use the distance between the center line and the center of the LR patch as the feature. The center line can be obtained from the horizontal projection of the image.
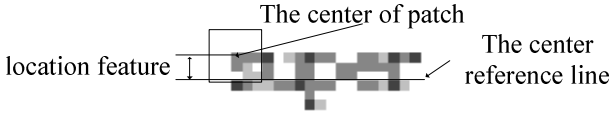
Fig. 4. The location feature

Every LR patch in the example database has record the distance as the attribute of this patch. The outlier is discriminated by the follow formulation:

$$outlier(x_i) = \begin{cases} false & if \ d(x_i, y_i) < t \\ true & other \end{cases}.$$ (6)

where t is the threshold, $d(\cdot, \cdot)$ is the distance of the attribute between the observe node y_i and candidates node x_i. In our work the threshold is 3 pixels which is just about quarter of the character height. The result of location pruning is shown in Fig.5(b).

● *The pruning with statistic feature.* Besides the location feature, some statistics are invariant or steady between the LR and HR patches. We use the gravity center as the feature. The gravity center of one LR patch lp is given in following:

$$xc = \frac{\iint x \cdot lpdxdy}{\iint lpdxdy}, \ yc = \frac{\iint y \cdot lpdxdy}{\iint lpdxdy}.$$ (7)

The gravity center is a measurement of the distribution of the density, so it is invariant by the degrade processing. The discrimination rule is similar as (6). In our work the threshold of the gravity center is the mean of the candidates in this node.

4   Experiments

To evaluate the performance of the proposed algorithm, the simulation LR image and the scanned image from the flat scanner will be demonstrated in the experiments. All experiments and timing statistics are carried out and recorded on executing the un-optimized code on the PC with Core 2 Duo 2.0G HZ CPU and 2G memory using the Matlab 2009.

4.1   The Training Example

In the previous works of text image SR, the patch pairs are directly extracted from the example text image without any special selection. Although this method is simple, the dataset will become vastness and tousle which is vital factor of the performance and complexity of the Markov network. Because of the fixed content of the text image, the train examples may select the special pattern to stand for the characters. We use the combination of all the character as the train example which is smallness, for example there are only 676 train image for all 26 English letter. The LR image is

degraded using the (5). The *PSF* is the average kernel and the noisy is the Gaussian white. The dataset contains only about $6*10^4$ patch pairs that are much less than the 10^6 of freeman's [3].
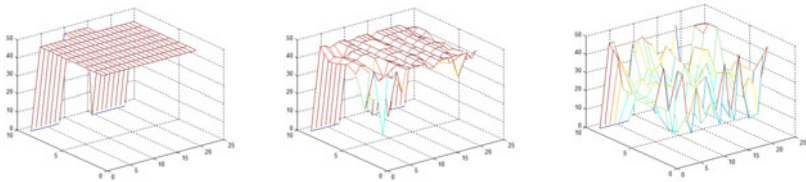
a) after background pruning   b) after location pruning   c) after statistic pruning

Fig. 5. The number of the candidates in every node. The position in the x-y plane is the location of the patches in the image, and the z axis gives number of the candidates.

4.2   The Simulation LR Image Experiment

This experiment uses 100 dpi. , 6 points, Time New Roman input images and shows the two times magnification factor of super resolved image. The input image is degraded by (8) with the average blur kernel from a true ground 200dpi. image. The Fig. 6 shows the result. The proposed results get higher PSNR than the result without pruning and the performance has improved.

a) LR input   b) truth HR   c) bicubic interpolate d) without pruning e) proposed method
                                     PSNR=6.62   PSNR=11.67   PSNR=12.01
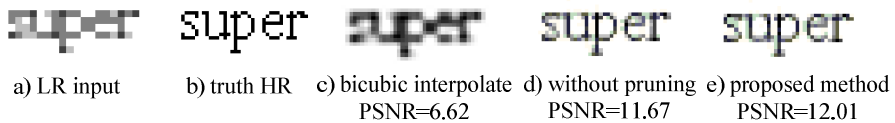
Fig. 6. The results of the simulation LR image. The d) is the same as the freeman's method except the database.

The performance of the different method is given in Table 1. The test dataset is random select one paragraph from an English examination paper, including 672 words. Because the performance has a negative relationship with the number of candidates, the computation time has been reduced since the outliers have been pruned. After structural pruning, there are only half candidates of the initial. Thus the construction Markov network will become rapid. As the number of nodes in the network is constant, the each iteration time is little change. However the iteration will convergent faster because the outlier is removed. Therefore the total time has been saved.

4.3   The Scanned Image Experiment

In this experiment the input image is scanned 6 points, Time New Roman document in 100dpi. The two times magnification super resolved image is shown in fig 7.

Table 1. The performance of simulation image

	Without pruning	Background pruning	structural pruning
Construction the MRF time (second)	359.81	255.18	**180.14**
Each iteration of BP time (second)	0.27	**0.25**	**0.25**
Mean of the candidates	50	33.14	**26.29**
Mean of Node number	588	588	588

a) the LR input   b) bicubic interploate   c) without pruning time =462.8s   d) after statistic pruning time=88.7s
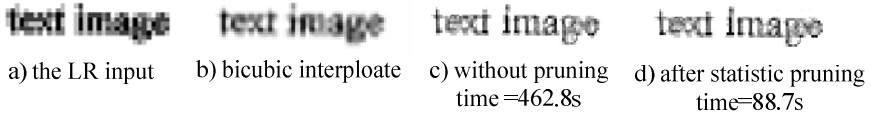
Fig. 7. The result of scanned image

5   Conclusion

We present a new neighbor constraint model for text image in Bayesian framework via the MRF. The outlier has been pruned based on the structure feature, so the complex of the network has decreased. A reduced example database will satisfy the demand of the network. From the demonstration of experiment, this method has better performance.

References

[1] Park, S.C., Park, M.K., Kang, M.G.: Super resolution image reconstruction: a technical overview. IEEE Signal Processing Magazine 20, 21–36 (2003)
[2] Baker, S., Kanade, T.: Limits on super resolution and how to break them. IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 1167–1183 (2002)
[3] Freeman, W.T., Jones, T.R., Paztor, E.C.: Example based super resolution. IEEE Computer Graphics and Applications 22(2), 56–65 (2002)
[4] Elad, M., Datsenko, D.: Example-Based Regularization Deployed to Super-Resolution Reconstruction of a Single Image. The Computer Journal 52(1), 15–30 (2009)
[5] Li, X., Lam, K.M., Qiu, G.P., Shen, L.S., Wang, S.Y.: An efficient example-based approach for image super resolution. In: IEEE Int. Conference Neural Networks & Signal Processing, pp. 575–580. IEEE Press, New York (2008)
[6] Yu-Wing, T., Liu, S., Brown, M.: Super Resolution using Edge Prior and Single Image Detail Synthesis. In: 28th IEEE Conference on Computer Vision and Pattern Recognition, pp. 2400–2407 (2010)
[7] Banerjee, J., Jawhar, C.V.: Super-resolution of Text Images Using Edge-Directedd Tangent Field. In: The Eighth IAPR workshop on Document Analysis Systems, pp. 76–83 (2008)

Integrating Local Features into Discriminative Graphlets for Scene Classification

Luming Zhang[1], Wei Bian[2], Mingli Song[1], Dacheng Tao[2], and Xiao Liu[1]

[1] Zhejiang Provincial Key Laboratory of Service Robot,
Computer Science College, Zhejiang University
{zglumg,brooksong,ender_liux}@cs.zju.edu.cn
[2] Centre for Quantum Computation and Information Systems,
University of Technology, Sydney
wei.bian@student.uts.edu.au, dacheng.tao@uts.edu.au

Abstract. Scene classification plays an important role in multimedia information retrieval. Since local features are robust to image transformation, they have been used extensively for scene classification. However, it is difficult to encode the spatial relations of local features in the classification process. To solve this problem, Geometric Local Features Integration(GLFI) is proposed. By segmenting a scene image into a set of regions, a so-called Region Adjacency Graph(RAG) is constructed to model their spatial relations. To measure the similarity of two RAGs, we select a few discriminative templates and then use them to extract the corresponding discriminative graphlets(connected subgraphs of an RAG). These discriminative graphlets are further integrated by a boosting strategy for scene classification. Experiments on five datasets validate the effectiveness of our GLFI.

Keywords: scene classification, graphlet, local features.

1 Introduction

Scene classification is an important issue for many multimedia applications, such as image retrieval and surveillance. To deal with scene classification successfully, it is essential to have proper discriminative image features. In the evolution of image analysis, many features have been proposed and they can be categorized into two groups: global features and local features. Global features, e.g., eigenspace [1], represent an image by a single vector and are hence tractable for conventional classifiers, such as Support Vector Machine(SVM) [13]. However, global features are sensitive to occlusions and clutters, which result in poor classification accuracy. In contrast to global features, local features, e.g., Scale Invariant Feature Transform(SIFT) [12], are extracted at interest points and are robust to image deformations. Different images may produce different number of local features. In order to be tractable for conventional classifiers, these local features are often integrated into an orderless bag-of-features representation. Unfortunately, as a non-structural representation, the bag-of-features representation ignores the spatial relations of local features, which prevents it from being discriminative.

To encode the spatial relations of local features for scene classification, graph based local feature integration [2–7] is proposed. In [2, 3], each image is modelled as a tree and image matching is formulated into tree matching. Unfortunately, compared to general graphs, the capability of modelling regions' spatial relations by trees is limited. Felzenszwalb et al. [4] modelled the relation of different parts of an object as a spring. However, [4] relies heavily on the optimal background subtraction. In [5], Hedau et al. defined a new measure of pairwise regions based on the overlaps between regions; but just region overlaps are too simple to capture the complicated spatial relations of regions. Keselaman et al. [6] defined a graph, called Least Common Abstraction (LCA), for an object. However, LCA cannot be output to a conventional classifier, e.g., SVM [13] directly. Walk kernel [7] captures the walk structures of regions by a finite sequence of neighboring regions. Unfortunately, as demonstrated in [8], the totter phenomenon brings noise to walk kernel [7] and thus makes it less discriminative.
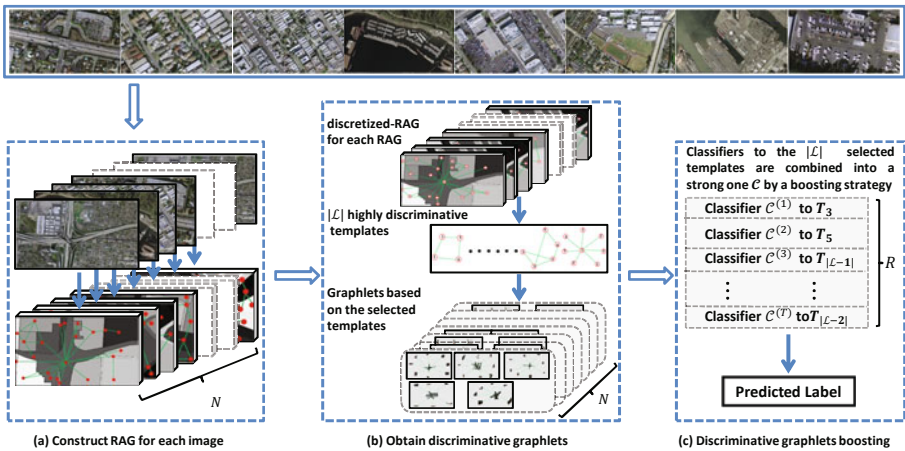
Fig. 1. The flowchart of our GLFI

To solve or at least reduce the aforementioned problems, a new local feature integration method GLFI is proposed for scene classification. As shown in Fig. 1, first of all, each scene image is segmented into a set of regions. To model the spatial relations of these regions, a graph called RAG is constructed subsequently (Fig.1(a)). Then, to measure a pair of RAGs, it is straightforward to compare all their pairwise graphlets. Unfortunately, based on graph theory, the number of graphlets of an RAG is huge, making the graphlet enumeration computational intractable. Towards an efficient measure, it is necessary to select a few discriminative graphlets for comparison. As the number of candidate graphlet for selection is huge, aiming at fewer candidates, we obtain templates by discretizing the continuous labels of graphlets into discretized ones, then only highly discriminative templates are selected and further used to extract the corresponding discriminative graphlets (Fig.1(b)). Finally, these discriminative graphlets are integrated by a boosting strategy for scene classification (Fig.1(c)).

2 Region Adjacency Graph(RAG)

A scene image usually contains millions of pixels. If we treat each pixel as a local feature, high computational cost will make scene classification computational intractable. Fortunately, a scene image can be represented by a set of clusters because pixels are usually highly correlated with their spatial neighbors, wherein each cluster consists of neighboring pixels with consistent color intensities. Thus, we propose RAG to represent a scene image by a set of regions and encode their spatial relations in a labelled graph.

(a). A scene image   (b). RAG   (c). Graphlet   (d). Template
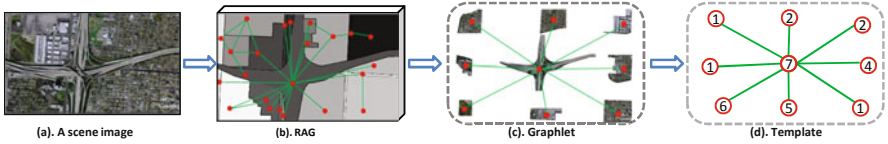
Fig. 2. The flowchart from a scene image(a) to its RAG(b) and further to its graphlet(c) and its template(d)

As shown in Fig.2(a, b), given a scene image I, we segment it into a set of regions $\{r_1, r_2, \cdots, r_M\}$ (Unsupervised Fuzzy Clustering(UFC) [17] based segmentation is applied because of its stability), and an RAG G is constructed to model a scene image I, i.e.,

$$G = (V, E, H, L, h, l) \qquad (1)$$

where $V = \{v_1, v_2, \cdots, v_M\}$ is a finite set of vertices, v_i represents region r_i; $h : V \rightarrow H$ is a function assigning a label to each $v \in V$, i.e., $h(v)$ is a row vector representing the RGB histogram of the region corresponding to v; $l : V \rightarrow L$ is a function assigning an index to each vertex $v \in V$, i.e., $l(v)$ means the region corresponding to v is obtained from the $l(v)$-th segmentation(multiple segmentations are applied); $E = \{(v_i, v_j)|v_i, v_j \in V \wedge l(v_i) = l(v_j) \wedge v_i \sim v_j\}$ is a set of edges, $v_i \sim v_j$ means two regions corresponding to v_i and v_j are spatial adjacent.

As shown in Fig.2(c), given an RAG G, we call S a graphlet of G if S is a connected subgraph of G. For two graphlets S and S', they are isomorphic [8], denoted by $S \cong S'$, if there exists a bijection $\varphi : V \rightarrow V'$ such that for each $u, v \in V, (u, v) \in E$ iff $(\varphi(u), \varphi(v)) \in E'$ and $h(u) = h(\varphi(u'))$. If $S \cong S'$ and $S' \subseteq G'$, we call S subgraph isomorphic to G' or, G' supergraph isomorphic to S, denoted by $S \sqsubseteq G'$.

3 Discriminative Graphlets Selection

Based on the definition of RAG, the similarity of a pair of scene images I and I' depends on their corresponding RAGs G and G'. To measure the similarity between G and G', it is straightforward to compare all their pairwise graphlets.

However, 1). based on graph theory, the number of graphlets of an RAG is $\mathcal{O}(M^M)$(usually $M > 50$); 2). non-discriminative graphlets make no contribution to scene classification. Therefore, it is necessary to select a few discriminative graphlets for scene classification.

Towards an efficient selection of discriminative grpahlets, a three-step method is developed: firstly, we obtain a small set of templates from the training RAGs and accordingly derive the class label of template. Then, a few discriminative templates are selected. Finally, these discriminative templates are used to extract the corresponding discriminative graphlets.

3.1   Template and Its Class Label

As shown in Fig.2(d), to obtain a template, a codebook $H^D = [h_1^D, h_2^D, \cdots, h_P^D]$ is generated by k-means [13] on all the training vertex labels firstly, then the continuous label $h(v)$ of vertex v is discretized into $h^D(v)$ by:

$$h^D(v) = \arg \min_{h \in H^D} ||h(v) - h|| \tag{2}$$

where $||\cdot||$ is the Euclidean norm. Based on H^D and h^D, given an graphlet S, we define its corresponding template T is obtained by mapping $g : S \to T$, where

$$T = \{V, E, H^D, L, h^D, l\} \tag{3}$$

Since template is a label-discretized graphlet, the number of candidate templates for selection is much smaller than that of graphlets, thus it is feasible to select a few discriminative ones for scene classification. Before selecting discriminative templates, we need to measure template's discrimination, i.e., how accuracy of a template predicting the class labels of scene images. As a label-discretized graphlet, template describes the spatial relations of local features in an approximate manner, to accurately predict the class label of template T, given an RAG G , it is necessary to find graphlets in G corresponding to T. Formally, we call graphlets S satisfying T, if $g(S) = T$, and graphlets of G satisfying T are collected into $G(T)$, i.e.,

$$G(T) = \{S | S \subseteq G \wedge g(S) = T\} \tag{4}$$

If $G(T) \neq \emptyset$, each graphlet $S \in G(T)$ can be represented as a vector $h(S)$, i.e.,

$$h(S) = \cup_{v \in S}[h(v)] \tag{5}$$

where $\cup[\cdot]$ is a row-wise vector concatenation operator.

Based on (5), given a set of training RAGs $\mathcal{G} = \{G_1, G_2, \cdots, G_N\}$ and a template T, we obtain a set of feature vectors $\mathcal{H} = \{h(S) | S \in G(T) \wedge G \in \mathcal{G}\}$, and further train a SVM classifier [13] $\mathcal{C}$ based on $\{\mathcal{H}, \mathcal{K}\}$, where $\mathcal{K}$ is the set of class labels corresponding to RAGs in $\mathcal{G}$. Based on trained SVM classifier $\mathcal{C}$, given an RAG G and a template T, the class label $k \in \{1, 2, \cdots, K\}$ of graphlet $S \in G(T)$ is obtained based on the posterior probability $P(G \to k|S)$ output from $\mathcal{C}$, i.e.,

$$S \to \arg \max_k P(G \to k|S) \tag{6}$$

Since there may be more than one graphlets in G satisfying template T, i.e., $|G(T)| \geq 1$, the label of $G(T)$ is derived from a multiple classifiers combining strategy [19] (under the sum rule), i.e., the posterior probability for $G(T)$ belonging to class $k \in \{1, 2, \cdots, K\}$ is:

$$P(G \to k|G(T)) = (1 - Z)P(G \to k) + \sum_{i=1}^{Z} P(G \to k|S_i) \tag{7}$$

where $Z = |G(T)|$; $P(G \to k)$ is the probability of RAG G belonging to class k (computed from the training RAGs), i.e.,

$$P(G \to k) = \frac{|G \to k \wedge G \in \mathcal{G}|}{N} \tag{8}$$

Based on (7), the class label of $G(T)$ is obtained by:

$$G(T) \to \begin{cases} \arg\max_k P(G \to k|G(T)) & \text{if } G(T) \neq \emptyset \\ 0 & \text{if } G(T) = \emptyset \end{cases} \tag{9}$$

where $G(T) \to 0$ means decision cannot be made on $G(T)$.

3.2   Selecting Discriminative Templates

In the extreme case, a template T is optimal if $\exists k \in \{1, 2, \cdots, K\}$, the following two conditions are satisfied:

C1: $P(G(T) \to k|G \to k) = 1$
C2: $P(G \to k|G(T) \to k) = 1$

where C1 maximize the descriptive ability of template T, and C2 maximize the discriminative ability of template T. However, as proved in [13], in the case of noisy training data, such optimal template may not always exist. Therefore, it is necessary to search for a set of sub-optimal templates, i.e., $\exists k \in \{1, 2, \cdots K\}$, such that:

C3: $P(G(T) \to k) \geq \min(P(G \to k))$
C4: $P(G \to k|G(T) \to k) \geq \alpha * P(G \to k)$

To satisfy C3, we obtain a set of discretized RAGs $\mathcal{G}^D = \{G^D|G^D = g(G) \wedge G \in \mathcal{G}\}$ based on (2), then the frequency of template T is computed by counting how many $G^D \in \mathcal{G}^D$ are supergraph isomorphic to T, i.e.,

$$P(G(T)) = \frac{|T \sqsubseteq G^D \wedge G^D \in \mathcal{G}^D|}{N} \tag{10}$$

Based on (10), the frequency of a template belonging to class $k \in \{1, 2, \cdots, K\}$ is computed by:

$$P(G(T) \to k) = \frac{|T \sqsubseteq G^D \wedge G^D \in \mathcal{G}^D \wedge G^D = g(G) \wedge G \to k|}{N} \tag{11}$$

For a template T, a larger $P(G(T) \to k)$ means T has a higher generalization ability towards class k. In our approach, an efficient frequent subgraph mining algorithm, FSG [18], is employed to output templates whose $P(G(T) \to k) \geq \min(P(G \to k))$

To satisfy C4, given a template T, its measure of discrimination is defined as largest discrimination towards class $k \in \{1, 2, \cdots, K\}$, i.e.,

$$disc(T) = \max_k \left[\frac{P(G \to k|G(T))}{P(G \to k)} \right] \tag{12}$$

where denominator is computed based on (8); the numerator is computed based on (7). Template whose $disc(T) < \alpha$ is regarded as a less discriminative one. Based on C3+C4, we present the algorithm of discriminative template selection in Table 1.

Table 1. Discriminative Template Selection(Algorithm 1)

---

input: A set of training data $\mathcal{D} = \{G_i, k_i\}_{i=1}^N$; Threshold α;
output: A set of discriminative template $\mathcal{L}$;
begin:
1. For each RAG G_i in $\mathcal{D}$, obtain the corresponding discretized-RAGs G_i^D and save them into $\mathcal{G}^D$;
2. Conduct FSG on $\mathcal{G}^D$ to output templates T whose $P(G(T) \to k) \geq \min(P(G \to k))$ into $\mathcal{L}$;
3. **for** each template $T \in \mathcal{L}$
    if $disc(T) < \alpha$, then $\mathcal{L} \leftarrow \mathcal{L} \setminus T$;
    end for;
Return $\mathcal{L}$;
end

3.3   Extracting Discriminative Graphlet

Each template $T \in \mathcal{L}$ (output from Algorithm 1) is discriminative. Thus given an input RAG G, we conduct depth-first-search on G, and graphlets of G satisfying T are extracted for scene classification. It is noticeable that, vertices in RAG are of low degree, i.e., less than 5 on average, so its computational is approximately linear increasing with the number of vertices in RAG G.

4   Discriminative Graphlets Boosting

To integrate the extracted discriminative graphlets for scene classification, a boosting strategy is developed. In detail, for each template $T \in \mathcal{L}$, a SVM classifier $\mathcal{C}$ is trained as described in Section 3.1. Based on $\{\mathcal{C}_i\}_{i=1}^{|\mathcal{L}|}$, we develop a multi-class boosting algorithm to integrate the $|\mathcal{L}|$ weak classifiers $\{\mathcal{C}_i\}_{i=1}^{|\mathcal{L}|}$ into a strong one $\mathcal{C}$. We present the algorithm of discriminative graphlets boosting in Table 2.

5   Experimental Results and Analysis

To demonstrate the advantage of our GLFI, we experiment on five datasets: Scene15 [9], Scene67 [20], Caltech256 [14], PASCAL VOC 2009 [15] and LHI [16]. Details of the five datasets are presented in Table 3.

Table 2. Discriminative Graphlets Boosting(Algorithm 2)

---

input: A set of training RAGs and their corresponding labels: $\{G_j, k_j\}_{j=1}^N$;
      A set of weak classifiers $\{\mathcal{C}_i\}_{i=1}^{|\mathcal{L}|}$; Iteration number of boosting R;
output: A strong classifier: $\boldsymbol{\mathcal{C}}(G)$;

---

begin:
1. Set the training RAG weights $w_j = \frac{1}{N}$, $j = 1, 2, \cdots, N$;
2. **for** $t = 1, 2, \cdots, R$
   (a).Select a weak classifier $\mathcal{C}^{(t)}$ from $\{\mathcal{C}_i\}$: $\arg\min_{\mathcal{C}^{(t)} \in \{\mathcal{C}_i\}} \sum_{j=1}^N w_j \cdot \prod(G_j(T_i) \nrightarrow k)$;
   (b).Compute weighted training error: $err^t = \frac{\sum_{j=1}^N w_j \cdot \prod(G_j(T_i) \nrightarrow k)}{\sum_{j=1}^N w_j}$;
   (c). $a^t \leftarrow \log \frac{(1-err^t)}{err^t} + \log(K - 1)$;
   (d). Update the training RAG weight: $w_j \leftarrow w_j \cdot \exp[a^t \cdot \prod(G_j(T) \nrightarrow k)]$;
   (e). Re-normalize w_j;
   end for;
Return $\boldsymbol{\mathcal{C}}(G) = \arg\max_k \sum_{t=1}^T a^t \cdot \prod(G(T_i) \rightarrow k)$;
end

Table 3. Details of the five datasets

Dataset	# of categories.	# of images.	# of training images	# of test images
Scene15	15	4485	100 per category	rest per category
Indoor67	67	15620	80 per category	20 per category
Caltech256	256	30,607	50 per category	rest per category
VOC2009	20	14,743	7,054	7,689
LHI	5	20	N/A	N/A
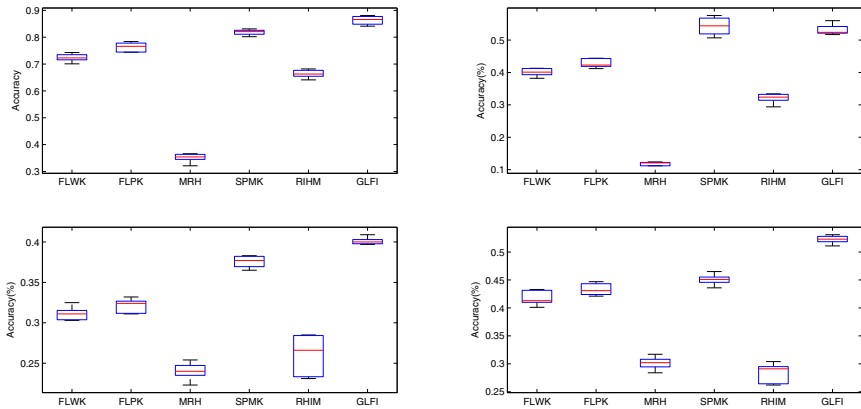
Fig. 3. Classification accuracy of the compared methods on Scene15(top left), Scene67(top right), Caltech256(bottom left) and PASCAL VOC 2009(bottom right)

5.1 GLFI versus Representative Local Features Integration Methods

In Fig. 3, we compare our GLFI with five representative local feature integration methods, i.e., fixed length walk kernel(FLWK) [7], fixed length path kernel

(FLPK) [8], multiresolution histogram(MRH) [10], spatial pyramid matching kernel(SPMK) [9] and region-based hierarchical image matching(RHIM) [2]. The experimental settings are as follows: the lengths of FLWK [7] and FLPK [8] are tuned from 2 to 10; for MRH [10], we smooth images with RBF kernels of 15 gray levels; for SPMK [9], each image is decomposed into over 1 million SIFT [12] features of 16×16 pixel patches computed over a grid with spacing of 8 pixels, then a codebook of size 400 is generated by k-means [13]; for our GLFI, the times of multiple segmentations, $\max(L)$, is tuned from 2 to 7, and the iteration number of boosting, R, is set to 200.

In Table 4, we present the classification accuracy of each category on PASCAL VOC 2009 . As seen, our GLFI outperforms the three compared graph based local feature integration methods significantly on most categories, which is consistent with our theoretical analysis in Section 1.

Table 4. Averaged classification accuracy of 20 categories on PASCAL VOC 2009(%)

	aero	bicycle	bird	boat	bottle	bus	car	cat	chair	cow
FLWK	72.2	40.6	41.2	42.1	23.9	56.6	39.8	44.3	47.2	20.2
FLPK	73.1	42.8	44.3	50.4	22.7	57.1	41.2	43.9	43.5	22.3
RHIM	60.8	22.1	25.3	33.2	11.3	34.6	30.1	26.3	30.2	13.2
GLFI	**75.6**	**54.1**	**60.6**	**58.2**	**33.4**	**65.2**	**56.5**	**56.4**	**48.5**	**37.7**
	dining	dog	horse	motor	person	potted	sheep	sofa	train	tv
FLWK	32.3	33.1	42.2	44.3	**76.6**	27.3	30.9	26.7	63.8	44.4
FLPK	33.7	34.4	44.5	44.5	73.2	**29.6**	**32.1**	28.4	65.3	46.7
RHIM	13.4	22.1	26.4	25.4	56.8	9.6	17.6	10.2	44.8	30.1
GLFI	**47.7**	**43.2**	**60.2**	**63.2**	74.6	29.4	31.3	**40.2**	**77.3**	**51.1**

5.2 Influence of Different Segmentation Settings

In retrospect to the proposed GLFI, we notice that the influence of segmentation operation in the construction of RAG is nonnegligible. To evaluate scene classification under different segmentation settings, based on (12), we report the frequent template's(output from Step2 of Algorithm 1) meausre of discrimination under benchmark-segmentation, deficient-segmentation, and over-segmentation. We experiment on PASCAL VOC 2009 [15] beacause its segmentation benchmark is helpful to make a precise comparison.

As shown in Fig. 4, templates from benchmark-segmentation achieves the highest discrimination, with the highest *disc* value of 35.4, followed by the over-segmentation 33.7 and deficient-segmentation 31.2. The explanations are as follows: 1).the benchmark segmentation is obtained by manually annotation, which encodes the high-level semantic understanding, thus it is unavoidable that UFC [17] may be less accurate than the benchmark segmentation; 2).in contrast with deficient-segmentation, more regions are obtained in over-segmentation setting, so it is rarer for one region spans several components, fewer discriminative components are neglected.
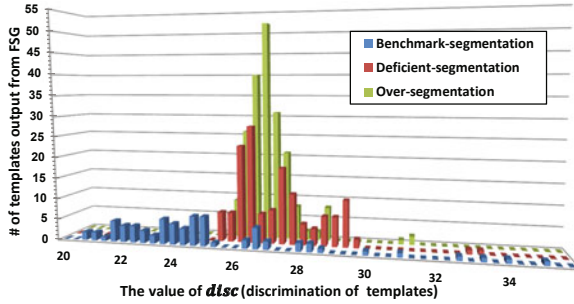
Fig. 4. disc value of templates under 3 different segmentation settings

5.3   Visualization of the Discriminative Graphlets

A unique property of our GLFI is the "transparency" of the scene classification model. As shown in Fig. 5, we visualize the the most discriminative graphlets of aerial images in LHI [16]. As seen, discriminative graphlets from different categories have different structure pattern, which further validates the intuition of our GLFI.

(a) Intersection

(b) Marine

(c) Parking

(d) Residential

(e) School
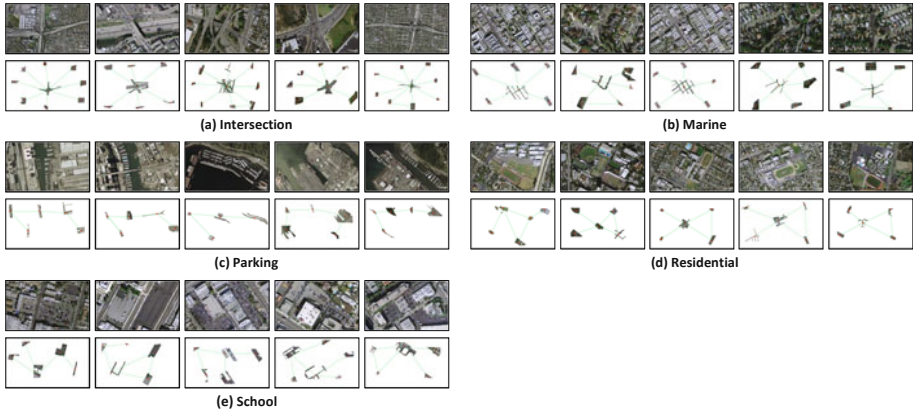
Fig. 5. Visualized discriminative graphlets

6   Conclusions

In this paper, a new local feature integration method GLFI is proposed for scene classification. First, an RAG is constructed to encode the geometric property and color intensity distribution of scene image. Then, the discriminative graphlets are selected from the RAGs. Finally, these discriminative graphlets are integrated by a boosting strategy for scene classification. Extensive experiments on five datasets validate the effectiveness of our GLFI.

References

1. Yuan, X., Zhu, H., Yang, S.: IEEE Workshop on Applications of Computer Vision and the IEEE Workshop on Motion and Video Computing, pp. 54–59 (2005)
2. Todorovic, S., Ahuja, N.: Region-based hierarchical image matching. IJCV (2007)
3. Demirci, et al.: Object recognition as many-to-many feature matching. IJCV 69(2) (2006)
4. Felzenszwalb, et al.: Pictorial structure for object recognition. IJCV 61(1) (2005)
5. Hedau, V., et al.: Matching images under unstable segmentations. In: CVPR (2008)
6. Keselman, Y., et al.: Generic Model Abstraction from Examples: TPAMI, 1141–1156 (2005)
7. Harchaoui, Z., Bach, F.: Image Classification with Segmentation Graph Kernels. In: CVPR, pp. 1–8 (2007)
8. Sherashidze, N., et al.: Efficient Graphlet Kernels for Large Graph Comparison. In: International Conference on Artificial Intelligence and Statistics, pp. 488–495 (2009)
9. Lazebnik, S., Schmid, C., Ponce, J.: Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In: ICCV, pp. 2169–2178 (2006)
10. Hadjidemetriou, E., et al.: Multiresolution Histograms and Their Use for Recognition. TPAMI, 831–847 (2004)
11. Cao, L., Fei-fei, L.: Spatially Coherent Latent Topic Model for Concurrent Segmentation and Classification of Objects and Scenes. In: ICCV, pp. 1–8 (2007)
12. Porway, J., Wang, K., Yao, B., Zhu, S.C.: Scale-invariant shape features for recognition of object categories, 90–96 (2004)
13. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley Interscience (2000)
14. Griffin, G., Holub, A., Perona, P.: (2007),
    http://authors.library.caltech.edu/769
15. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: (2009),
    http://www.pascal-network.org/challenges/VOC/voc2009/workshop/index.html
16. Yao, B., Yang, X., Zhu, S.-C.: Introduction to a Large Scale General Purpose Ground Truth Dataset: Methodology, Annotation Tool, and Benchmarks. In: Yuille, A.L., Zhu, S.-C., Cremers, D., Wang, Y. (eds.) EMMCVPR 2007. LNCS, vol. 4679, pp. 169–183. Springer, Heidelberg (2007)
17. Xiong, X., Chan, K.L.: Towards An Unsupervised Optimal Fuzzy Clustering Algorithm for Image Database Organization. In: ICPR (2000)
18. Kuramochi, M., Karypis, G.: An Efficient Algorithm for Discovering Frequent Subgraphs. TKDE, 1038–1051 (2004)
19. Kittler, J., Hatef, M., Duin, R.P.W., Matas, J.: On Combining Classifier. TPAMI, 226–239 (1998)
20. Quattoni, A., Torralba, A.: Recognizing indoor scenes. In: CVPR (2009)

Opponent and Feedback:
Visual Attention Captured

Senlin Wang[1], Mingli Song[1], Dacheng Tao[2], Luming Zhang[1],
Jiajun Bu[1], and Chun Chen[1]

[1] Zhejiang Provincial Key Laboratory of Service Robot,
Computer Science College,
Zhejiang University, Hangzhou 310027, China
{snail_wang,brooksong,zglumg,bjj,chenc}@zju.edu.cn
[2] Centre for Quantum Computation and Information Systems,
University of Technology, Sydney
dacheng.tao@gmail.com

Abstract. Visual attention, as an important issue in computer vision
field, has been raised for decades. And many approaches mainly based on
the bottom-up or top-down computing models have been put forward to
solve this problem. In this paper, we propose a new and effective saliency
model which considers the inner opponent relationship of the image in-
formation. Inspired by the opponent and feedback mechanism in human
perceptive learning, firstly, some opponent models are proposed based
on the analysis of original color image information. Secondly, as both
positive and negative feedbacks can be learned from the opponent mod-
els, we construct the saliency map according to the optimal combination
of these feedbacks by using the least square regression with constraints
method. Experimental results indicate that our model achieves a better
performance both in the simple and complex nature scenes.

Keywords: Visual attention, Opponent, Feedback, Saliency map.

1   Introduction

Visual attention influences our daily behaviors as if an invisible hand behind. It
plays an important role in many applications such as image/video processing,
visual advertisement design and human psychology. For example, in image and
video compression, using the visual attention model permits a rapid selection
for image information before further processing [1, 2]. It is also widely used in
image segmentation [3], classification [4] and retrieval [5]. What's more, seizing
people's attention is also important in advertising design [6].

Current visual attention methods can be divided into three models based on
the different driving conditions. Stimulus-driven model [7–9] is computed from
a set of low-level features. Some features such as color, intensity and orientation
are extracted from the original image at different scales and orientations. The
stimulus-driven model can perform well for some nature scenes or synthetic data.
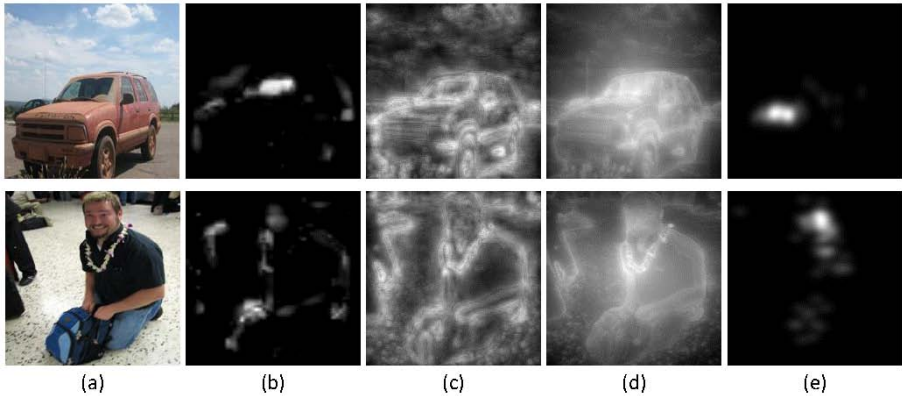
Fig. 1. Comparison on some existing saliency models and eye-tracking data. (a) Original color images (b) Itti et al. saliency maps [7] (c) Torralba et al. saliency maps [9] (d) Judd et al. saliency maps [10] (e) Eye-tracking data.

However, for some of the nature scenes with complex scenario, they could not predict the right place where human look. Fig. 1(b) is the saliency map generated by Itti et al. model [7] which was computed from color, intensity and orientation features. And Fig. 1(c) is the Torralba et al. saliency map [9] based on the subband pyramids feature. There is a big difference between the saliency maps and the real eye-tracking data in Fig. 1(e). The eye-tracking data are from the MIT eye-tracking data set [10] which consists of 1003 images from 15 viewers.

Besides low-level features, some high-level features like face, human and other objects [10, 11] were also added to the stimulus-driven model for a better results. This method is treated as the concept-driven model. Cerf et al. [11] added face detection into Itti et al. stimulus-driven model and improved the saliency map's accuracy significantly. Judd et al. [10] expanded the model further, which included not only high-level features but also mid-level features. Then they trained a SVM model from the eye-tracking data set to learn different feature's parameter for the final saliency map. However, these methods ignored the inner relationship among different features. Fig. 1(d) takes Judd et al. saliency model as an example, and the brightest areas of the map can not match the eye-tracking data well.

The information-driven [12, 13] model also makes a contribution to the visual attention issue. Bruce et al. [12] believed that the saliency region provided more information than others, and a method called "Attention based on Information Maximization (AIM)" was proposed to maximize the self-information in the image. It performed marginally better than the previous models. Except for these three common driven models mentioned before, other models such as Bayesian model [14, 15], efficient coding [16], and spectral residual [17] provided some different methods for the topic as well.
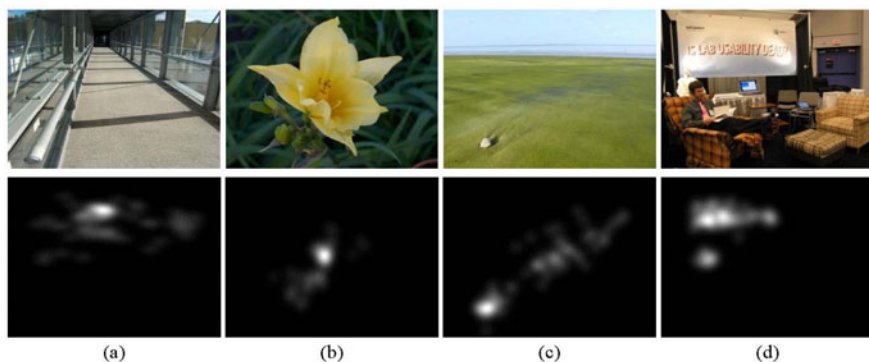
Fig. 2. Some typical images reflecting different opponent process. The first row is the original images while the second row is the eye-tracking data. (a) Intensity Opponent (b) Center Opponent (c) Self-information Opponent (d) Semantic Opponent.

In this paper, inspired by the opponent and feedback (OAF) mechanism in human perceptive learning, we learn different opponent relationship from the image and use some effective models to reflect them firstly. Then, feedbacks provided by the opponent models are studied and a new saliency model can be constructed by using the least square regression with constraints method.

The rest of the paper is organized as follows: section 2 introduces the motivation of the proposed approach. Our saliency model based on the OAF mechanism is described in section 3. Experiments and analysis are carried out in section 4. And we make a conclusion in section 5.

2   Motivation

Opponent-process theory [18] is a universal psychological and neurological model in human perceptive learning. The well-known application is the color-opponent process. Color-opponent process [19] states that human visual system responses color information by strengthening one color and at the same time suppressing another color. This opponent mechanism is meaningful in the analysis of visual attention issue and has been already involved in many saliency models [7, 10, 11].

In addition to color-opponent mechanism in human visual system, we believe that some other opponent mechanisms also exist when people look at an image. Fig. 2 shows some typical images and their eye-tracking data. And it's clear that some other opponent mechanisms also play an important role in both simple and complex nature scenes.

Firstly, the intensity-opponent mechanism is taken into consideration. It is a part of the color-opponent mechanism [19] while popular color-opponent only includes two color pairs, blue versus yellow, and red versus green. Secondly, as people naturally take the significant object in the center of the images when they take photos (Fig. 2(b)), the center region in the image is more significant
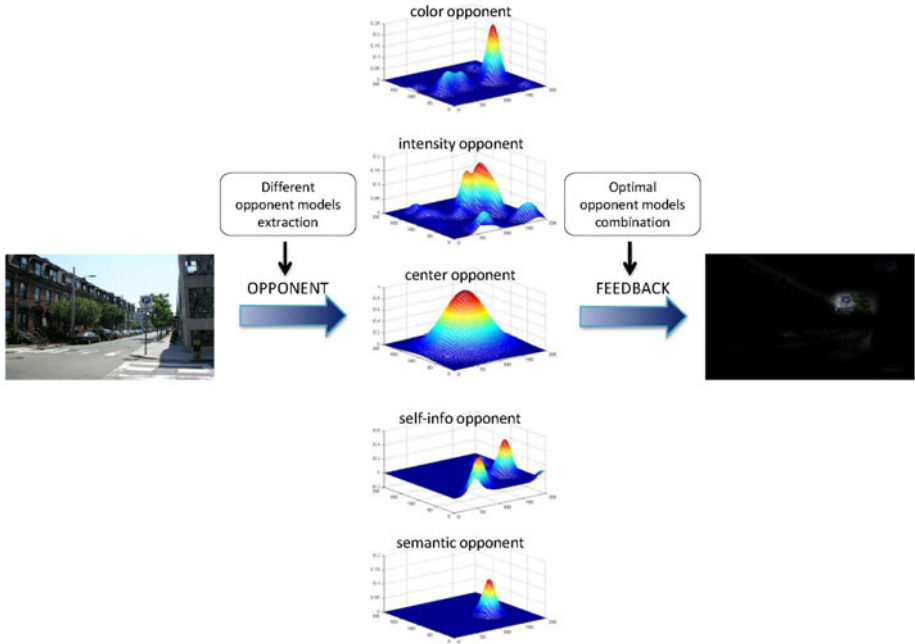
Fig. 3. Opponent and Feedback (OAF) mechanism

than others. And we put forward a center-opponent mechanism to strengthen the center area and suppress the surroundings. Thirdly, visual salience is not only based on the response of cells within a local region but also on the relationship between the local region and the surroundings [12]. And the bigger difference will attract more attention than the smaller ones. It can be confirmed in Fig. 2(c) where the white stone on the grass is more attractive than other parts. Finally, semantic-opponent relationship is also learned in our saliency model. When many different semantic objects appear in one image simultaneously, only one or two objects are saliency. For example, in Fig. 2(d), only text and face information are noticed by human.

3   Saliency Model Based on OAF

In this section, we present the saliency model which is motivated by the opponent and feedback (OAF) mechanism in human perceptive learning. Fig. 3 gives a graphical overview of the OAF mechanism and more details are depicted in the following parts.

3.1   Opponent Process

As described in section 2, besides the classical color opponent process in human perceptive learning, some other opponent mechanisms also play an important

role when human look at an image. And we construct five typical opponent models to learn the saliency model in this section.

Color Opponent and Intensity Opponent. Color opponent relationship exists in the pairs of red/green, blue/yellow in human primary visual cortex [20], and intensity opponent mechanism can detect light/dark variations from the image and learn their opponent relationship. Here Itti's method [7] is used to extract the two opponent maps from original images at different scales and orientations.

Center Opponent. Center information is another important factor that affects human's attention. In order to simulate the center-opponent process, we learn a normal Bivariate Gaussian equation (1) which has been proved to fit the eye fixation distribution well for the center bias [15].

$$M(x,y) = \eta \exp\Big(-\frac{1}{2(1-\rho^2)}\Big(\frac{(x-\mu_1)^2}{\delta_1^2} \quad +2\rho\frac{(x-\mu_1)(y-\mu_2)}{\delta_1\delta_2} + \frac{(y-\mu_2)^2}{\delta_2^2}\Big)\Big) \tag{1}$$

M is the image matrix and x, y are the point coordinates, $\eta = \frac{1}{2\pi\delta_1\delta_2\sqrt{1-\rho^2}}$ and we use $\rho = 0$ for simplicity. For the image size of 200*200 in this paper, the specific parameters are $\mu_1 = 98.1, \mu_2 = 102.6, \delta_1 = 35.6, \delta_2 = 40.1$.

Self-Information Opponent. It's naturally for human to notice the area that is different from surroundings. And self-information is a measure of the information content associated with others. The maximum self-information implies the biggest difference with others in the image. Therefore, we adopt a model proposed in [12], in which the maximum self-information can be extracted from the image. And in order to achieve a better opponent effect, only top 15% points are selected from the generating map.

Semantic Opponent. Let's take Fig. 2(d) as an example, there is a man who sits on the sofa reading a book, and around him is some furniture, two notebooks and a curtain with some words. The corresponding eye-tracking image shows that the text and face information attract more attention rather than the sofa or notebooks. This phenomenon expresses that the opponent relationship also exists among different semantic objects. However, it's quite difficult to decide which object is more salient as not all objects appear simultaneously in one image and their scales, positions can not be the same either.

Here three common semantic features (text, face and car) are extracted from the image to learn the opponent relationship, and a statistical average score method is used to decide the salient one. Firstly, images with at least two of these three features are selected from the data set. Then we compute each feature's saliency average score denoted as S_k using (2). For an image with three features, the most salient feature's score is three, and the others are zero because of the opponent effect. The same process is applied for the image with any two features

| (a) | (b) | (c) | (d) |

Fig. 4. Difference between combination with equal and optimal opponent models. (a) Original image (b) Modulated image with equal combination (c) Modulated image with optimal combination (d) Modulated image of eye-tracking. For a clearly contrast, we adopt top 15% points of the saliency maps.

and the more salient one gets two points. In (2), $score_i$ is the feature's score in the image i, and N_k is the number of the images with feature k, $k \in \{1, 2, 3\}$ is the class label for text, face and car.

$$S_k = \frac{1}{N_k} \sum_{i=1}^{N_k} score_i \tag{2}$$

For the 1003 images in MIT data set, 665 images have at least one of these three features and 123 images satisfy our statistical standard. The scores are 1.94, 0.67, 0.18 for text, face and car, which means text information attracts more attention than face and car. The "winner-take-all" mechanism [21] is used to ensure that only the salient feature remains and others are suppressed. It's noticeable that even in the images without high-level features, our saliency model can still get good results due to the existence of other significant opponent mechanisms.

3.2   Feedback Process

The feedback process is an optimal linear combination of different opponent maps. The strengthened parts from the opponent maps provide positive feedbacks to the original image while the suppressed parts provide negative feedbacks. We use the linear, least square regression with constraints [22] to learn their weights w from the real data. Formally, F is the set of all the opponent maps which are stretched to vectors.

$F = \{color, intensity, center, self\text{-}information, semantic\}$.

While G_{fix} is the vectorized fixation map which represented as the fixation locations convolved with a gaussian kernel. And we use (3) to learn the different weights for each opponent model.

$$\arg \quad \min_{w}(||F * w - G_{fix}||)^2 \qquad s.t. \quad w \geq 0 \tag{3}$$

The equation can be treated as a LASSO problem where efficient implementation exists [23]. The optimal weights we learned are 0.279, 0.203, 0.135, 0.463 and

| (a) Original | (b) Itti [7] | (c) Judd [10] | (d) Bruce [12] | (e) Hou [17] | (f) Our | (g) Eyetrack |
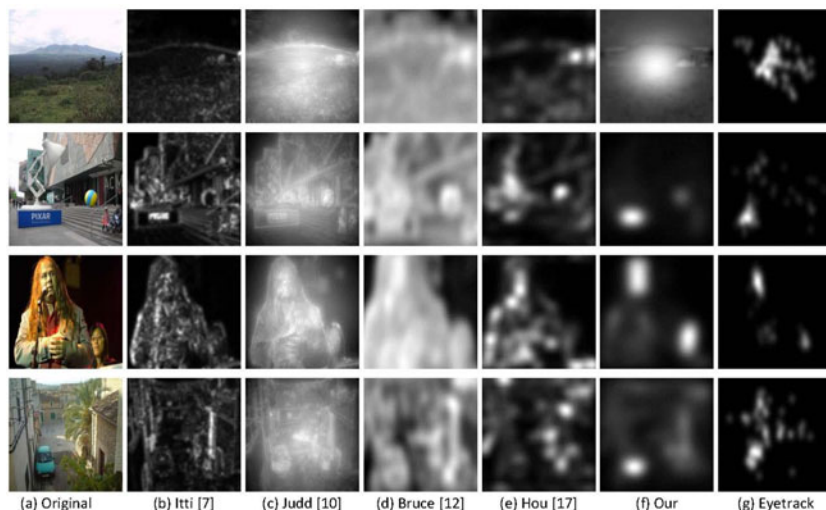
Fig. 5. Qualitative comparison on different saliency models

0.627 which correspond to each variable in F. Then we normalize the saliency map for the final result. Simple as it is, however, experimental results show a good performance.

4 Experiments and Analysis

Experiments are implemented on the MIT data set [10] which is the largest data set and records eye-tracking data from 15 users viewing 1003 images freely. All the images are resized to 200*200 before further processing, and both qualitative and quantitative analysis are carried out here.

4.1 Qualitatively Evaluation

The first column in Fig. 5 is the original color images and the corresponding eye-tracking data are presented in the last column. We compare our results to four typical saliency models, stimulus-driven model(Fig. 5(b)), concept-driven model (Fig. 5(c)), information-driven model(Fig. 5(d)) and spectral residual model (Fig. 5(e)). Our OAF model is presented in Fig. 5(f). The saliency area is the brightest regions in the saliency maps. As the OAF mechanism used in our model, the saliency areas in the results are relative concentration and match the eye-tracking data better than other methods in both simple and complex nature scenes.

4.2 Quantitatively Evaluation

The receiver operating characteristic (ROC) curve is a widely accepted criterion to evaluate the saliency model quantitatively [10, 15]. And the area under the
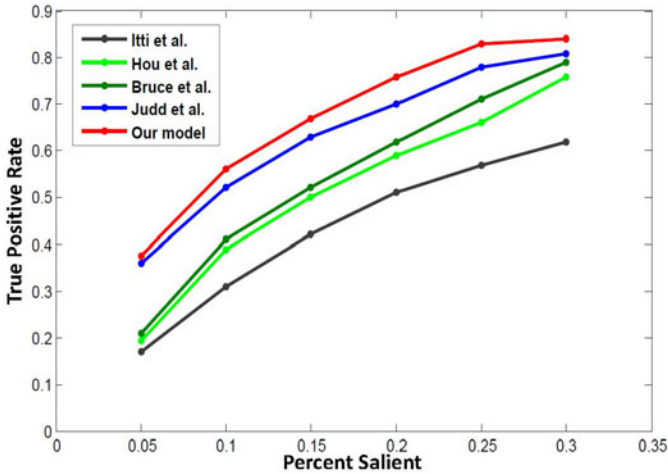
Fig. 6. The *ROC* curves of five different models

curve indicates how well the saliency map predicting eye fixations. In Fig. 6, we plot our result curve as well as the other saliency result curves. The threshold of visual attention region varies from top 5% to top 30% part of the saliency map. The *ROC* curves in the figure depict quantitatively that the proposed method takes advantage over the other methods.

5   Conclusion

In this paper, some representative opponent models are extracted from the original image firstly. Then we learn the optimal combination of different models by using the least square regression algorithm. Though simple, our *OAF* model takes opponent relationship among different features into consideration and produces better results than several existing typical models.

We also notice some limitations of our method. For example, more high-level features can lead a better opponent relationship in some images. In the future, we will discuss the relationship among more high-level features, and better feedback process will be also studied.

References

1. Bradley, A.P., Stentiford, W.M.: Visual Attention for Region of Interest Coding in JPEG 2000. J. Vis. Commun. Image Represent. 14, 232–250 (2003)
2. Itti, L.: Automatic Foveation for Video Compression Using a Neurobiological Model of Visual Attention. IEEE Transactions on Image Processing 13, 1304–1318 (2004)

3. Ramanathan, S., Katti, H., Sebe, N., Kankanhalli, M., Chua, T.-S.: An Eye Fixation Database for Saliency Detection in Images. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6314, pp. 30–43. Springer, Heidelberg (2010)

4. Siagian, C., Itti, L.: Rapid Biologically-inspired Scene Classification Using Features Shared with Visual Attention. IEEE Transactions on Pattern Analysis and Machine Intelligence 29, 300–312 (2007)

5. Marques, O., Mayron, L.M., Borba, G.B., Gamba, H.R.: Using Visual Attention to Extract Regions of Interest in the Context of Image Retrieval. In: ACM Southeast Regional Conference, pp. 638–643 (2006)

6. Pieters, R., Wedel, M.: Attention Capture and Transfer in Advertising Brand Pictorial and Text-size Effects. Journal of Marketing 68, 36–50 (2004)

7. Itti, L., Koch, C., Niebur, E.: A Model of Saliency-based Visual Attention for Rapid Scene Analysis. Pattern Analysis and Machine Intelligence 20, 1254–1259 (1998)

8. Meur, O.L., Callet, P.L., Barba, D., Thoreau, D.: A Coherent Computational Approach to Model Bottom-up Visual Attention. Pattern Analysis and Machine Intelligence 28, 802–817 (2006)

9. Oliva, A., Torralba, A.: Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope. International Journal of Computer Vision 42, 145–175 (2001)

10. Judd, T., Ehinger, K., Durand, F., Torralba, A.: Learning to Predict Where Humans Look. In: International Conference on Computre Vision, pp. 2106–2113 (2009)

11. Cerf, M., Harel, J., Einhauser, W., Koch, C.: Predicting Human Gaze Using Low-level Saliency Combined with Face Detection. Neural Information Processing Systems (2007)

12. Bruce, N.D.B., Tsotsos, J.K.: Saliency, Attention, and Visual Search: An Information Theoretic Approach. Journal of Vision 9, 1–24 (2009)

13. Zhang, L., Tong, M.H., Cottrell, G.W.: SUNDay: Saliency Using Natural Statistics for Dynamic Analysis of Scenes. In: Annual Cognitive Science Society Conference (2009)

14. Itti, L., Baldi, P.: Bayesian Surprise Attracts Human Attention. In: Neural Information Processing Systems (2005)

15. Yang, Y., Song, M., Li, N., Bu, J., Chen, C.: What is the Chance of Happening: A New Way to Predict Where People Look. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6315, pp. 631–643. Springer, Heidelberg (2010)

16. Vig, E., Dorr, M., Barth, E.: Efficient Visual Coding and the Predictability of Eye Movements on Natural Movies. Spatial Vision 22, 397–408 (2009)

17. Hou, X., Zhang, L.: Saliency Detection: A Spectral Residual Approach. Computer Vision and Pattern Recognition 1, 1–8 (2007)

18. Solomon, R.L.: The Opponent-process Theory of Acquired Motivation: the Costs of Pleasure and the Benefits of Pain. American Psychologist 35, 691–712 (1980)

19. Foster, M.: A Text-book of Physiology. Lea Bros.&Co., London (1891)

20. Engel, S., Zhang, X., Wandell, B.: Colour Tuning in Human Visual Cortex Measured with Functional Magnetic Resonance Imaging. Nature 388, 68–71 (1997)

21. Koch, C., Ullman, S.: Shifts in Selective Visual Attention: Towards the Underlying Neural Circuitry. Human Neurobiology 4, 219–227 (1985)

22. Zhao, Q., Koch, C.: Learning A Saliency Map Using Fixated Locations in Natual Scenes. Journal of Vision 11, 1–15 (2011)

23. Tibshirani, R.: Regression Shrinkage and Selection via the Lasso. Journal of the Royal Statistical Society 58, 267–288 (1996)

Depth from Defocus
via Discriminative Metric Learning

Qiufeng Wu[1,2], Kuanquan Wang[1], Wangmeng Zuo[1], and Yanjun Chen[1]

[1] School of Computer Science and Technology,
Harbin Institute of Technology, Harbin, 150001, China
[2] College of Science, Northeast Agricultural University,
Harbin, 150030, China
neauqfwu@gmail.com

Abstract. In this paper, we propose a discriminative learning-based method for recovering the depth of a scene from multiple defocused images. The proposed method consists of a discriminative learning phase and a depth estimation phase. In the discriminative learning phase, we formalize depth from defocus (DFD) as a multi-class classification problem which can be solved by learning the discriminative metrics from the synthetic training set by minimizing a criterion function. To enhance the discriminative and generalization performance of the learned metrics, the criterion takes both within-class and between-class variations into account, and incorporates margin constraints. In the depth estimation phase, for each pixel, we compute the N discriminative functions and determine the depth level according to the minimum discriminant value. Experimental results on synthetic and real images show the effectiveness of our method in providing a reliable estimation of the depth of a scene.

Keywords: Depth estimation, Discriminative learning, Discriminative metric, Sub-gradient descent.

1   Introduction

Depth from defocus (DFD) is to infer the depth of each point in the scene from multiple defocused images which are captured with different camera settings. Compared with other image-based depth estimation approaches, e.g., depth from stereo and structure from motion, DFD can avoid correspondence problems [1].

Since the introduction of DFD for depth estimation [2], many DFD approaches, e.g., Markov random field-based approach [3], spatial domain-based approach [4], partial differential equation-based approach [5, 6], and variational approach [7–9], have been developed. All these approaches are faced with the problem of choosing the appropriate Point Spread Function (PSF) (e.g., Gaussian function and Pillbox function) in the imaging process. However, bypassing this problem, Favaro et al. [10, 11] develop a learning-based approach, which repeatedly learns the linear operators from a training set of blurred images of a certain depth using singular value decomposition (SVD) for a number of depths. Nevertheless, there still exist some weaknesses for this learning-based approach. For example, learning the linear operator of a certain depth is only from the

training set of blurred images of the same depth using SVD, but it does not consider the influence of the training sets of blurred images of other depths. This possibly leads a new image of a certain depth to belong to the null spaces of the linear operators of other depths.

In this paper, we proposed a novel approach based on discriminative learning, which consists of a discriminative learning phase and a depth estimation phase. For simplicity, we assume that the surface of the scene is equifocal and that dimension of admissible depth level (class) set $S = \{s_1, s_2, \cdots, s_N\}$ is finite. Based on this hypothesis, DFD is formalized as multi-class classification. In the discriminative learning phase, the results of [10] and [11] are extended to learn the discriminative functions and the synthetic training set are used to learn N discriminative metrics (the values of parameters) in N discriminative functions by minimizing a criterion function, which is solved by the sub-gradient descent method [12]. In the depth estimation phase, we compute the values of the N discriminative functions and select the class label (depth level) corresponding to the minimum discriminant value.

Once the discriminative metrics are learned via labeled training set, they can be used to estimate the depth of the various scenes in depth estimation phase. Since the whole computation only involves simple matrix-vector multiplication operations and can be performed independently at each pixel, allowing for a high level of parallelism, the algorithm can be performed efficiently. The experimental results on synthetic and real images show that the depth map of the scene is significantly enhanced with greater accuracy and robustness over the state-of-the-art method.

2 Related Work

Discriminative learning approaches have recently received much attention, which have been used in face identification [13], 3D scan data segmentation [14] and image annotation[15] etc.. In [11], learning approach is used in DFD. Favaro and Soatto[11] select the equifocal imaging model and the least-squares criterion to formulate DFD as an optimal inference. In the Hilbert space, the least-squares criterion is stated as:

$$\phi(s, r) = \|\mathrm{I} - H_s r\|^2 \tag{1}$$

where s and r is the depth and radiance of the scene respectively, $H_s : L^2(\mathbb{R}^2) \to \mathbb{R}^P$ is the linear operator such that $H_s r \dot{=} \langle \mathrm{h}^s(\cdot, \mathrm{y}), r \rangle = \int \mathrm{h}_\mathrm{v}^s(\mathrm{y}, \mathrm{x}) r(\mathrm{x}) d\mathrm{x}$, the observed image $\mathrm{I} = [I_1; I_2; \cdots; I_K] \in \mathbb{R}^P$ is a column vector of dimension $P = MNK$ by stacking K images of the same scene captured with K different parameter settings $\mathrm{v} = [v_1, v_2, \cdots, v_K]^T$ on top of each other, $\mathrm{h}_v^s = [h_{v_1}^s, h_{v_2}^s, \cdots, h_{v_K}^s]^T$ is the PSF corresponding to each parameter setting.

Using the geometry of operators in Hilbert spaces, Favaro et al. [11] proved that the extremum of (1) is identical to the extremum of the following function:

$$\psi(s) = \|H_s^\perp \mathrm{I}\|^2 \tag{2}$$

where $H_s^\perp$ is the orthogonal operator.

If the surface is constrained to an N dimensional set of admissible depth levels and the orthogonal operators $H_s^\perp$ corresponding to each depth level are known, the corresponding discriminative function for each depth level is determined. As a result, for each point in the measured images, the class (depth level) corresponding to the smallest discriminant value is selected by computing (2) (the discriminative functions).

Favaro et al. [11] learn the orthogonal operator $H_s^\perp$ for each depth level from training set by SVD. For more detailed description of learning procedure please refer to [11]. A basic limitation of using SVD to obtain the orthogonal operators is that $H_{\hat{s}}^\perp$ obtained by SVD at a given depth level $\hat{s}$ can only guarantee that the value of $\|H_{\hat{s}}^\perp I_{\hat{s}}\|^2$ for the defocused images $I_{\hat{s}}$ at given depth level $\hat{s}$ is small, but can not guarantee that the value of $\|H_{\hat{s}}^\perp I_{\hat{s}}\|^2$ is smaller than the value of $\|H_{\tilde{s}}^\perp I_{\hat{s}}\|^2$ at other depth level $\tilde{s}$. Therefore, it is easier to determine the depth level for the defocused images $I_{\hat{s}}$ to be $\tilde{s}$. To avoid this scenario, it is proposed to learn the discriminative metrics by minimizing the criterion function that takes both within-class and between-class variations into account.

3 Learning Discriminative Metric for DFD

In this section, we first present the core of model for discriminative metric learning, and then describe the sub-gradient descent algorithm for learning the discriminative metric, and finally provide the procedure for depth estimation.

3.1 Problem Formalization

As discussed in Section 1 and Section 2, DFD can be formalized as multi-class classification. Using the finite dimension of admissible depth level set $S = \{s_1, s_2, \cdots, s_N\}$ as the number of category, we extend (2) to represent the discriminative function for each depth level. Therefore, using notation of Euclidean and Mahalanobis distance, the N discriminative functions can be written as:

$$g_{s_i}(I) = \|L_i I\|^2 = I^T M_i I, i = 1, 2, \cdots, N \qquad (3)$$

where I is an unclassified sample image that is a column vector of dimension P (as is shown in Section 2), L_i is a matrix of size $P \times P$, the matrix $M_i = L_i^T L_i$. As L_i and M_i appear in discriminative function, L_i and M_i are called *discriminative metric* and *Mahalanobis discriminative metric* at depth level s_i respectively. For all admissible depths, if $\mathbf{L} = (L_1, L_2, \cdots, L_N)$ or $\mathbf{M} = (M_1, M_2, \cdots, M_N)$ is known, the class of I corresponding to the smallest discriminant value is selected by computing (3).

The subsequent step shows how to reconstruct the criterion function and learn the N discriminative metrics for DFD. According to (3), for arbitrary training sample I_{ij} (which denotes the *jth* training sample at *ith* depth level s_i), if I_{ij} is classified into the depth level s_i correctly, the value of $\|L_i I_{ij}\|^2$ must be smaller than the values of $\|L_k I_{ij}\|^2 (\forall k \neq i)$, that is, $\|L_i I_{ij}\|^2 \leq \|L_k I_{ij}\|^2 (\forall k \neq i)$.

Moreover, to increase the robustness of classification, the large margin constraint is incorporated such as $\|L_k I_{ij}\|^2 \geq \|L_i I_{ij}\|^2 + 1 (\forall k \neq i)$. Obviously, if $(\|L_i I_{ij}\|^2 + 1) - \|L_k I_{ij}\|^2 (\forall k \neq i, \forall i, j)$ are smaller than zero, a robust and perfect classification accuracy would be obtained. Therefore, to obtain higher classification accuracy, the N discriminative metrics can be obtained by minimizing the following criterion function:

$$\sum_{i,j,k} (1 - y_{ik})[1 + \|L_i I_{ij}\|^2 - \|L_k I_{ij}\|^2]_+ \tag{4}$$

where the term $[z]_+ = max(z, 0)$, a variable $y_{ik} = 1$ is introduced if and only if $i = k$; otherwise $y_{ik} = 0$.

Additionally, by Favaro's work [11], the training set at any depth level should be the null space of the corresponding discriminative metric. In other words, for any training sample I_{ij} in the training set at the depth level s_i, the value of $\|L_i I_{ij}\|^2$ should be as small as possible. Therefore, we further introduce a term into (4) and obtain the following criterion function:

$$\varepsilon(\mathbf{L}) = (1 - \mu) \sum_{i,j} \|L_i I_{ij}\|^2 + \mu \sum_{i,j,k} (1 - y_{ik})[1 + \|L_i I_{ij}\|^2 - \|L_k I_{ij}\|^2]_+ \tag{5}$$

where $\mu \in [0, 1]$ is balance parameter trading off the effect between two terms.

The criterion function in (5) is multi-modal and non-convex in the matrix elements of $\mathbf{L}$. To avoid these difficulties, we reformulate the optimization of (5) as a semi-definite programming (SDP). We firstly convert (5) into the criterion function with respect to $\mathbf{M}$:

$$\varepsilon(\mathbf{M}) = (1 - \mu) \sum_{i,j} I_{ij}^T M_i I_{ij} + \mu \sum_{i,j,k} (1 - y_{ik})[1 + I_{ij}^T M_i I_{ij} - I_{ij}^T M_k I_{ij}]_+ \tag{6}$$

Furthermore, we convert (6) into a SDP by introducing nonnegative slack variables $\{\xi_{ijk}\}$ for all triplets which do not satisfy the large margin constraint. The SDP is stated as:

$$\min(1 - \mu) \sum_{i,j} I_{ij}^T M_i I_{ij} + \mu \sum_{i,j,k} (1 - y_{ik}) \xi_{ijk} \tag{7}$$

$$s.t. \begin{cases} I_{ij}^T M_k I_{ij} - I_{ij}^T M_i I_{ij} \geq 1 - \xi_{ijk} \\ \xi_{ijk} \geq 0 \\ M_i \succeq 0 \end{cases} . \tag{8}$$

3.2 Sub-gradient Descent

The core of learning the discriminative metrics is to minimize the criterion in (6). Here we adopt the sub-gradient descent method [12], providing that it converges to the correct solution. In the sub-gradient descent method, $\mathbf{M}$ iterates along the gradient of $\varepsilon(\mathbf{M})$ in (6) with respect to $\mathbf{M}$ to reduce the criterion function and then projects this result onto the cone of all positive semi-definite matrices

at each iteration, which converges to a matrix $\mathbf{M}'$ while the gradient step-size is sufficiently small.

The core of the sub-gradient descent method is gradient computation. Suppose $C_{ij} = I_{ij}I_{ij}^T$ and define the set of triples Φ^t and Ψ^t respectively. Both $(i, j, k) \in \Phi^t$ and $(i, j, k) \in \Psi^t$ result in triggering the second term of (5) and (6) by the indices (i, j, k). But notice the fact that $(i, j, k) \in \Phi^t$ implies that the large margin constraint $1 + \|L_i I_{ij}\|^2 \leq \|L_k I_{ij}\|^2$ does not hold, while $(i, j, k) \in \Psi^t$ implies that the large margin constraint $1 + \|L_k I_{kj}\|^2 \leq \|L_i I_{kj}\|^2$ does not hold. With those definitions, at tth iteration, we can write the gradient of $\varepsilon(\mathbf{M}^t)$ with respect to M_i^t as:

$$\frac{\partial \varepsilon}{\partial M_i^t} = (1 - \mu) \sum_j C_{ij} + \mu(\sum_{i,j,k \in \Phi^t} C_{ij} - \sum_{i,j,k \in \Psi^t} C_{kj}) \tag{9}$$

To accelerate the sub-gradient descent method, results by SVD are used as a starting point for the SDP. For more detailed description of the sub-gradient descent method please refer to [12].

3.3   Recovery of Depth

After $\mathbf{L}$ or $\mathbf{M}$ is obtained, (3) is used to estimate the depth information of the scene. Given a collection of K defocused images of the scene, for each pixel x, the patch centered in x for each defocused image is extracted, and then a column vector $I(x) = [I_1(x); I_2(x); \cdots; I_K(x)]$ is obtained by stacking them on top of each other. We compute the N discriminative functions according to (3). At each pixel, the depth level s corresponding to the minimum among the values of the discriminative functions is the actual depth level s^*:

$$s^*(x) = \arg\min_{s_i \in S} g_{s_i}(I(x)) \tag{10}$$

4   Experimental Results

4.1   Experiments with Synthetic Images

In the discriminative learning phase, the 51 synthetic scenes are obtained by placing a synthetic scene of 100×100 pixels obtained randomly equidistantly in the range of $520mm$ and $850mm$ in front of a camera with a $35mm$ lens and F-number 4. For each scene, two defocused images are captured by bringing the plane at $520mm$ and $850mm$ into focus respectively. 196 patches of 7×7 pixels are collected from each defocused images, which constitute the training set with 196×51 labeled samples. The discriminative metrics learned from this training set are used on two synthetic testing sets to obtain Fig. 1 and Fig. 2.

Fig. 1 shows the depth estimation performance when we use the discriminative metrics learned from training test with ranks of $45, 60, 70, 80$ and 90 by SVD and the proposed algorithm respectively on synthetic images. Both mean and
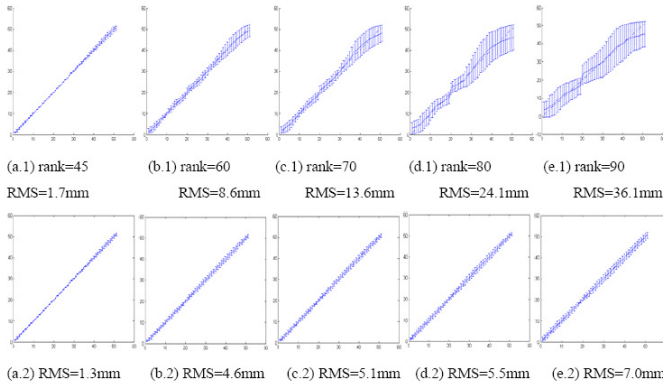
Fig. 1. Performance test for SVD and the proposed algorithm. Top (a.1-e.1): Results using the orthogonal operators from training set with the truncated ranks of 45, 60, 70, 80 and 90 by SVD on the testing set. Bottom (a.2-e.2): Results using the discriminative metrics from training set by the proposed algorithm on testing set. Both mean and standard deviation of the estimated depth are plotted over the ideal curve.

standard deviation of the estimated depth (solid line) are plotted over the ideal characteristic curve (the dotted line). Notice that, for SVD, when the chosen rank does not correspond to the correct rank of the matrices, the performance degrades rapidly and the value of the mean error (RMS) varies from $1.7mm$ to $36.1mm$. However, for the proposed algorithm, this case is not worried and the RMS varies only from $1.3mm$ to $7.0mm$. Therefore, the rank without consideration is one of the most important merits.

To estimate further performance of the proposed algorithm, we synthesize a set of two novel images of 51×2601 pixels that are segmented in horizontal stripes of 51×51 pixels (see Fig. 2). Every stripe has been generated by the same random radiance but with equifocal planes at decreasing depths as we move from top to the bottom of the image. Fig. 2 shows that depth map and mesh of the reconstructed depths are obtained by the proposed algorithm.

4.2   Experiment with Real Images

In the experiment with real data, the training set consists of 196×51 labeled samples, which are captured by the camera with a $12mm$ lens and F-number 2 . The far and near focused plane is identical to Section 4.1. The discriminative metrics learned from this training set are used on Fig. 3(a) to obtain Fig. 3(b). Fig. 3 shows the two defocused images and the depth map before and after median filtering.

The discriminative metrics learned by SVD and our proposed algorithm in Section 4.1 are used on Fig. 4(a) to obtain Fig. 4(b) and Fig. 4(c) respectively. Fig. 4 shows that the estimated depth map by SVD is very similar to the estimates obtained by our proposed algorithm.
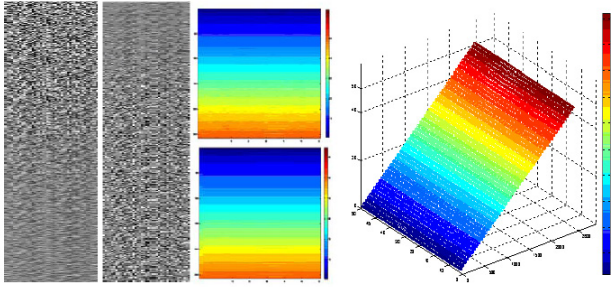
Fig. 2. Performance test for the proposed algorithm with synthetic data. Left: two synthetic defocused images are shown. The surface in the scene is a piecewise constant function (a stair) such that each horizontal stripe of 51×51 pixels corresponds to an equifocal plane. Depth levels decrease moving from the top to the bottom of the images. Middle: the depth map of the stair scene before and after a 7×7 pixels median filtering respectively. Right: mesh of the reconstructed depth of the stair scene.
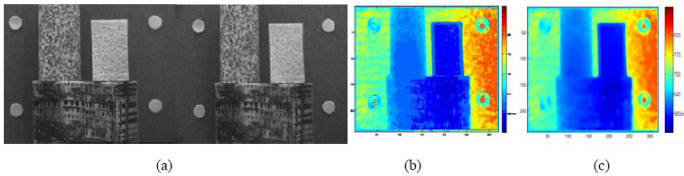
Fig. 3. (a) Detail of two 240×320 defocused images. For more details on the scene and camera settings, please refer to [6]; (b) and (c) Depth map recovered by the proposed algorithm before and after post-processing respectively.
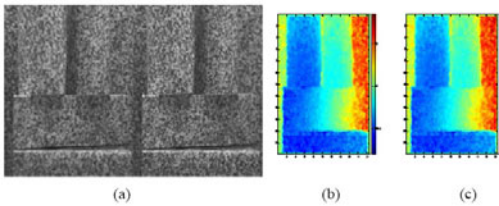
Fig. 4. (a) Detail of two 238×205 defocused images. For more details on the scene and camera settings, please refer to [11]; (b) and (c) Depth map recovered by SVD and the our proposed algorithm respectively.

5   Conclusion

We present a novel method for recovering the depth of a scene from multiple defocused images, which consists of a discriminative learning phase and a depth estimation phase. Experimental results on synthetic and real images demonstrate effectiveness of our method in providing a reliable estimate of the depth

of a scene. The method is robust, and the discriminative metrics learned from synthetic defocused images can be used effectively on real images. The method only involves simple matrix-vector multiplication computations in the depth estimation phase and can be performed independently at each pixel, and thus the algorithm is high level of parallelism and real-time performance.

References

1. Schechner, Y.Y., Kiryati, N.: Depth from defocus vs. stereo: how different really are they? Int. J. Comput. Vision 39(1), 141–162 (2000)
2. Pentland, A.: A new sense for depth of field. IEEE Trans. Pattern Anal. Mach. Intell. 9(4), 523–531 (1987)
3. Rajagopalan, A.N., Chaudhuri, S.: An MRF model-based approach to simultaneous recovery of depth and restoration from defocused images. IEEE Trans. Pattern Anal. Mach. Intell. 21(7), 577–589 (1999)
4. Ziou, D., Deschenes, F.: Depth from defocus estimation in spatial domain. Computer Vision and Image Understanding 81, 143–165 (2001)
5. Namboodiri, V.P., Chaudhuri, S.: On defocus, diffusion and depth estimation. Pattern Recognition Letters 28, 311–319 (2007)
6. Favaro, P., Soatto, S., Burger, M., Osher, S.J.: Shape from defocus via diffusion. IEEE Trans. Pattern Anal. Mach. Intell. 30(3), 518–531 (2008)
7. Rajagopalan, A.N., Chaudhuri, S.: A variational approach to recovering depth from defocused images. IEEE Trans. Pattern Anal. Mach. Intell. 19(10), 1158–1164 (1997)
8. Jin, H., Favaro, P.: A Variational Approach to Shape from Defocus. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2351, pp. 18–30. Springer, Heidelberg (2002)
9. Favaro, P.: Recovering thin structures via nonlocal-means regularization with application to depth from defocus. In: CVPR, pp. 1133–1140 (2010)
10. Favaro, P., Soatto, S.: Learning Shape form Defocus. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2351, pp. 735–745. Springer, Heidelberg (2002)
11. Favaro, P., Soatto, S.: A geometric approach to shape from defocus. IEEE Trans. Pattern Anal. Mach. Intell. 27(3), 406–417 (2005)
12. Weinberger, K.Q., Saul, L.K.: Distance metric learning for large margin nearest neighbor classification. Journal of Machine Learning Research 10, 207–244 (2009)
13. Guillaumin, M., Verbeek, J., Schmid, C.: Is that you? Metric Learning approaches for face identification. In: ICCV, pp. 498–505 (2009)
14. Anguelov, D., Taskar, B., Chatalbashev, V.: Discriminative learning of Markov Random Fields for segmentation of 3D scan data. In: CVPR, pp. 169–176 (2005)
15. Guillaumin, M., Mensink, T., Verbeek, J., Schmid, C.: TagProp: Discriminative metric learning in nearest neighbour models for image auto-annotation. In: ICCV, pp. 309–316 (2009)

Analysis of the Proton Mediated Feedback Signals in the Outer Plexiform Layer of Goldfish Retina

Nilton Liuji Kamiji[1,2], Masahiro Yamada[1], Kazunori Yamamoto[1], Hajime Hirasawa[1], Makoto Kurokawa[2], and Shiro Usui[1]

[1] Laboratory for Neuroinformatics, RIKEN Brain Science Institute,
2-1 Hirosawa, Wako, Saitama 351-0198, Japan
[2] Department of Biological Sciences, Tokyo Metropolitan University,
Hachioji, Tokyo 192-0397, Japan
{nilton,myamada,yamkaz,hajime_hirasawa}@brain.riken.jp,
kurokawa-makoto@tmu.ac.jp, usuishiro@riken.jp

Abstract. Center-surround antagonistic receptive field in the retina is generated by negative feedback from horizontal cells (HCs) via a proton feedback mechanism [1]. In this study, the contribution of protons on the color opponent signal formation is analyzed. Increasing the buffer capacity of the external medium by 10 mM HEPES depolarized the dark membrane potential of HCs, and substantially increased hyperpolarizing responses to light stimulation. In contrast, feedback mediated depolarizing responses of H2 and H3 HCs were suppressed by HEPES. Moreover, depolarizing response onset of H2 and H3 HCs was significantly delayed compared to the hyperpolarizing responses. These indicate that proton plays an important role on the color opponent signal formation of HCs, and that the feedback from H1 to H2 HCS is delayed by 10 – 20 ms. A similar delay might be applicable to other feedback pathways as well.

Keywords: Retina, outer plexiform layer, horizontal cell, proton, feedback.

1 Introduction

Negative feedback signals from horizontal cells (HCs) to cones contribute to the center-surround antagonistic receptive field formation and color signal transformation in lower vertebrates. Mechanisms of dynamic characteristics of color signaling, however, have not been understood well. Cone and rod photoreceptor signals are segregated at HC level in fish [2-3]. H1, H2 and H3 HCs are cone-driven HCs, receiving major input from red-, green- and blue-sensitive cones, respectively [4].

Stell et al. [4] proposed morphologically that H1 HCs send back their negative feedback signals to all types of cones. This negative feedback signals have long been believed to be mediated by GABAergic feedback from HCs to cones. This GABAergic feedback hypothesis has been doubted by the fact that picrotoxin, a GABA receptor antagonist, could not suppress the feedback signal [5]. Proton (H^+) was proposed as a feedback transmitter released by HC depolarization instead of GABA [1]. Acidification of extracellular environment reduces the current through voltage gated calcium channels (I_{Ca}) and shifts the activation voltage to more positive

voltages [1], [6-7]. I_{Ca} of cone photoreceptors was also suppressed when HCs were depolarized during darkness or by application of glutamate agonists [1]. The effects of HC polarisation on the I_{Ca} of cone photoreceptors disappeared when the retina was superfused with a high pH-buffer [1], [8]. Based on these observations, Hirasawa and Kaneko [1] hypothesised that protons released from HCs regulate the I_{Ca} of cone terminals and hence the transmitter release. Vessey et al. [9] made similar observations in the zebrafish retina by Ca^{2+} imaging studies of cone terminals. Proton release from fish HCs was revealed by ratio imaging method for measurement of immediate extracellular surface pH of HCs and the H^+ release was found to be dependent on the membrane depolarization of HCs [10]. The H^+ releasing mechanism was found to be due to V-ATPase (vacuolar type adenosine triphosphatase, H^+ pump) based on the results that bafilomycin A1, a specific V-ATPase inhibitor, suppressed acidification responses induced by kainate or high extracellular K^+ [10].

Mechanisms of dynamic characteristics of color signaling were studied on the basis of the proton negative feedback hypothesis. HEPES and/or chromatic adapting light were applied to analyze the effects on spectral responses of H1, H2 and H3 HCs. Chromatic response kinetics and response amplitude were investigated to examine the proton mediated feedback in chromatic signaling of these HCs.

2 Material and Methods

2.1 Retinal Preparation and Intracellular Recording

Goldfish (Carassius auratus) of about 30-50 g body weight were dark-adapted for 5 min, pithed and dissected. The experimental procedures conformed to the Guidelines of the Institute Animal Welfare Committee. The eye balls were removed under dim red light and the retina was isolated and mounted with photoreceptor-side up on a piece of white filter paper (AAWP01300, Japan Millipore Ltd., Japan). This preparation is referred to here as 'light-adapted retina' since the fish were dark-adapted for only 5 min before surgery. The retina was placed in a small recording chamber (volume ~0.3 ml) and superfused at a rate of 0.4 ml/min with control bicarbonate-Ringer's solution, having the following composition (in mM): 102 NaC1, 2.6 KC1, 28 NaHCO$_3$, 1.0 MgCl$_2$, 1.0 CaC1$_2$ and 5.0 glucose. The bicarbonate-Ringer was saturated with a gas mixture of 95 % O$_2$ and 5 % CO$_2$, maintaining a pH of 7.60 at room temperature (20 – 24 ℃). In experiments using 10 mM HEPES supplemented Ringer, the same amount of NaCl was replaced by HEPES. The pH values of the solutions were strictly maintained within 0.01 pH unit fluctuation. Intracellular recordings were made using a glass-microelectrode filled with 3 M KC1 (resistances, 100 - 400 MΩ). The pH buffering capacity of the 10 mM HEPES-fortified Ringer was calculated to be 9.1 mM, which is 2.7 times greater than that of the control bicarbonate-Ringer (3.4 mM) [11].

Responses of three types of cone-driven HCs were intracellularly recorded. The cell somata were identified by (i) the position of the electrode tip at around 100 - 150 μm from the surface of the photoreceptor layer, (ii) characteristic spectral responses of each type (by comparing responses to five chromatic test stimuli), (iii) the

sequential order of the encountered photoresponses, and finally (iv) morphological identification by the intracellular staining of Lucifer yellow, in some cases.

2.3   Light Stimulation

The photostimulator for test light stimuli consisted of five types of LEDs (461, 524 584, 614 and 689 nm; reflection type with parallel light emission, LAP-type, Iwasaki Electric Ltd., Tokyo), corresponding to blue, green, yellow, orange and red light respectively. Their light intensities at the retinal surface were adjusted to equal quanta of 4.6×10^4 quanta/μm^2s, each equipped with an interference filter (10 nm narrow band width). The test light intensity induced light response around 10-20 mV at the impalement of HCs. Each test light of five color stimuli was flashed with 700 ms duration at 2.1 s intervals in a cycle of 11 s. The five sets of monochromatic light stimuli were used for the identification of cone-driven HCs.

2.4   Estimation of Response Onset Delays

Light responses of HCs are delayed due to phototransduction delay at the cone photoreceptors, and the synaptic transmission delay from cones to HCs. To a first approximation, the initial phase of a HC light response was fitted by the following second order system equation

$$G(s) = Ae^{-ds}\big/\left(s + 1/\tau\right)^2 ,\qquad (1)$$

where A is the gain, τ the time constant, and d the time delay. More accurate fits can be obtained by increasing the order of the equation; however, they also introduce system delays. Therefore, a second order equation was utilized in the present study. Parameters A and τ were fitted to the initial response phase while delay time d was globally searched in the range from 10 ms to 80 ms in 1 ms step.

3   Results

3.1   Effects of HEPES on Dark Membrane Potential and Photoresponse Amplitude of H1 HCs

H1 HCs were intracellularly recorded from goldfish retinae which were dark-adapted for only five minutes. Responses to five chromatic light stimuli perfused with control Ringer's solution were compared with those perfused with 10 mM HEPES-fortified Ringer. By the application of HEPES containing Ringer, the dark membrane potential of H1 HCs was significantly depolarized (Fig. 1A). On average, it depolarised by 10.7 ± 1.7 mV from -39.2 ± 2.4 mV to -29.5 ± 2.8 mV (p value of paired Student's t-test =0.0003, n=15). Light response amplitudes of H1 HCs to all the five chromatic light stimuli were significantly increased by more than 50 % by HEPES (Figs. 1B). These effects were suggested to be attributed to suppression of proton mediated feedback from HCs to cones. The delay on the photoresponse onset determined by fitting the initial phase of the response to a second order system (1) (Fig. 1C) was not significantly different between different stimulus wavelengths in control condition,
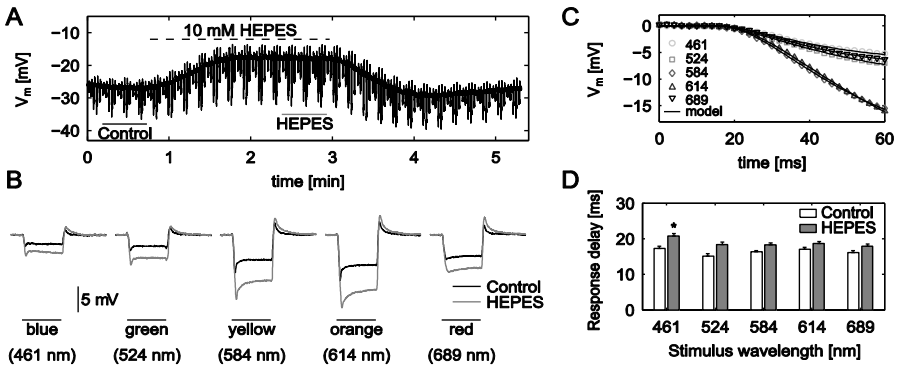
Fig. 1. Effects of HEPES on H1 HC response to 5 color light stimuli. A) Continuous recordings to sequential light stimulation. Dashed line: range of drug application; straight lines: range of averaged trials for analysis. B) Averaged responses of three trials in control (black lines) and HEPES (gray lines) condition at different stimulus wavelength. Horizontal bars: light stimulation (700 ms). C) Initial phase of an H1 HC's responses (symbols) fitted by a second order system with delay (solid lines). D) Averaged response onset delay at each stimulus wavelength obtained as in C (mean ± SE, n = 13). *: Responses in HEPES were significantly delayed from those in control. Blue responses were significantly different from remaining wavelengths in HEPES.

however, response onset delay in HEPES was significantly delayed compared to that in control condition (Fig. 1D). The delay being on average by 4.1 ± 0.2 ms to blue; 3.1 ± 0.2 ms to green; 2.1 ± 0.1 ms to yellow; 2.0 ± 0.1 ms to orange; and 1.1 ± 0.2 ms to red (paired Student's t-test: p=0.0007, 0.002, 0.009, 0.041 and 0.026, respectively, n=13). And blue responses were significantly delayed compared from other wavelengths in HEPES condition (Fig. 1D; one way anova, p=0.01, n=13).

3.2 Effects of HEPES on Dark Membrane Potential and Photoresponse Amplitude of H2 HCs

By the application of 10 mM HEPES Ringer, the dark membrane potential of H2 HCs was significantly depolarized (Fig. 2A). On average, it depolarised by 11.6 ± 1.8 mV from -38.5 ± 3.9 mV to -26.9 ± 2.7 mV (paired Student's t-test: p=0.00003, n=13). Depolarizing red (689 nm) light responses of the cells were greatly suppressed by HEPES (Fig. 2B), on average the suppression of response amplitude being 2.1 ± 0.5 mV from 3.7 ± 0.6 mV to 1.5 ± 0.3 mV (Δ=-55 ± 6 %, paired Student's t-test: p=0.001, n=13). On the other hand, hyperpolarising light response amplitudes of H2 HCs to short-middle wavelengths were enhanced by the application of HEPES (Fig. 2B). These effects were suggested to be attributed to suppression of proton mediated feedback from HCs to cones. The initiation of photoresponse onset was not affected by HEPES (Fig. 2C & D). However, depolarizing photoresponse onset for red light stimulation was significantly delayed compared to other stimulus wavelengths by 14.0 ± 2.9 ms to orange and by 17.0 ± 2.6 ms to green (one way anova: p=0.00002) in control condition. That delay became greater by 20.5 ± 6.5 ms to orange and by 25.7 ± 4.5 ms to green (p=0.0002) in HEPES condition (n=3).
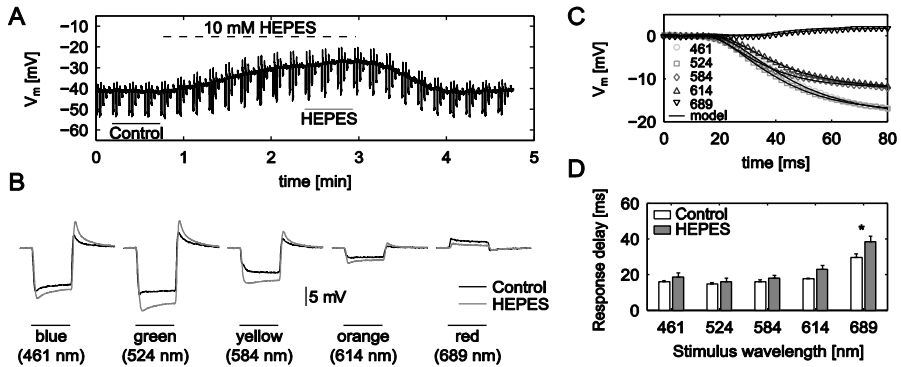
Fig. 2. Effects of HEPES on H2 HC response to 5 color light stimuli. A) Continuous recordings to sequential light stimulation. Dashed line: range of drug application; straight lines: range of averaged trials for analysis. B) Averaged response of three trials in control (black lines) and HEPES (gray lines) condition. Horizontal bars: light stimulation (700 ms), numbers indicate applied wavelength. C) Initial phase of an H2 HC's responses (symbols) fitted by a second order system (solid lines). D) Response onset delay at each stimulus wavelength (mean ± SE, n=3); *: significantly different from remaining condition.

3.3   Effects of HEPES on Dark Membrane Potential and Photoresponse Amplitude of H3 HCs

By the application of 10 mM HEPES Ringer, the dark membrane potential of H3 HCs was also significantly depolarized (Fig. 3A). On average, it depolarised by 12.4 ± 1.9 mV from -28.8 ± 3.2 mV to -16.4 ± 3.8 mV (paired Student's t-test: p=0.0001, n=10). Hyperpolarizing blue light responses were significantly enhanced by HEPES (Figs. 3B), on average the enhancement of response amplitude being 3.8 ± 0.7 mV from 8.6 ± 2.1 mV to 12.8 ± 1.7 mV (Δ=80 ± 30 %, p=0.0003, n=10). As the wavelength of green light stimulus is near the neutral wavelength of H3 HCs where its response polarity changes, the green response waveforms looked very complicated due to the summation of two kinds of inputs with opposite polarity from green- and blue-sensitive cones. The green response was usually composed of an early hyperpolarising potential followed by a late depolarising one. The green responses measured at the early hyperpolarising peak were significantly enhanced by HEPES (Fig. 3B), on average the enhancement of response amplitude being 2.3 ± 0.8 mV from 5.7 ± 1.1 mV to 8.0 ± 1.5 mV (Δ=93 ± 52 %, p=0.023, n=10) (Fig. 3B). Depolarizing yellow and orange light responses were greatly suppressed by HEPES (Fig. 3B). On average, the suppression of response amplitude was 10.0 ± 1.8 mV to yellow from 12.8 ± 1.7 mV to 2.7 ± 1.4 mV (Δ= -81 ± 9 %, p=0.0004, n=10) and 9.0 ± 1.7 mV to orange from 10.8 ± 1.6 mV to 1.9 ± 0.9 mV (Δ= -83 ± 7 %, p=0.0005, n=10), respectively (Fig. 3B). Hyperpolarising red responses were also significantly suppressed by HEPES (Fig. 3B), on average the suppression of response amplitude being 2.1 ± 0.5 mV from -2.2 ± 0.6 mV to -0.1 ± 0.3 mV (Δ= -117 ± 25 %, p=0.002, n=10) (Fig. 3B). Depolarizing photoresponse onset for yellow and orange light stimulation were significantly delayed by 20.2 ± 3.5 ms and 23.6 ± 3.2 ms, respectively when compared to hyperpolarizing photoresponses onset to blue light in
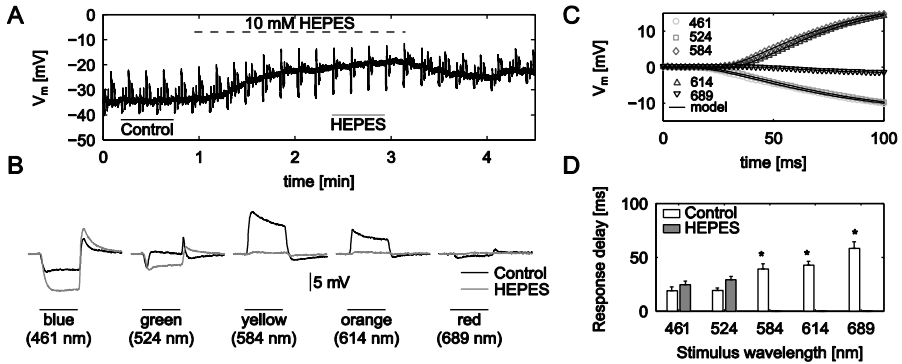
Fig. 3. Effects of HEPES on H3 HC response to 5 color light stimuli. A) Continuous recordings to sequential light stimulation. Dashed lines: drug application; straight lines: range of averaged trials for analysis. B) Averaged response of three trials in control (black lines) and HEPES (gray lines) condition. Horizontal bars: light stimulation (700 ms). C) Initial phase of an H3 HC's responses (symbols) fitted by a second order system (solid lines). D) Response onset delay at each stimulus wavelength (mean ± SE, n = 5). *: Red responses were significantly delayed from yellow, and those to yellow and orange were significantly delayed from blue and green. Responses in HEPES were also significantly delayed from those in control.

control condition (one way anova, p=0.000006, n=5). The initiation of photoresponse onset was also not affected by HEPES (Fig.3C & D) for blue and green stimuli. For other stimulus wavelengths HEPES almost completely suppressed response amplitude, therefore response onset delay could not be measured.

4   Discussion

In the present study, we demonstrated that HEPES depolarized the dark membrane potential of HCs and increased hyperpolarizing responses. These observations are consistent with previous studies where bicarbonate buffer was completely substituted by HEPES [15]. In addition, HEPES also suppressed the depolarizing responses in H2 and H3 HCs, demonstrating the contribution of the proton feedback mechanism on the color opponent signal formation of H2 and H3 HCs. A schematic diagram of the connection between cones and HCs is illustrated in Fig. 4. Straight lines indicate glumatatergic synapses, whereas dashed lines indicate proton feedback synapses.

The HEPES-induced depolarization of the dark membrane potential of HCs indicates that, the synaptic cleft is maintained in a slightly acidified state in darkness by proton extruded from depolarized HCs via V-ATPase proton pumps [10] together with proton exocytosed with glutamate release [16-17]. Light stimulation decreases glutamate release by hyperpolarizing the cones, which also hyperpolarizes HCs (solid lines in Fig. 4). These hyperpolarizations induce alkalinization of the synaptic cleft (dashed lines in Fig. 4), eliciting an increase in calcium current at the cone synaptic terminal [6], therefore, augmenting glutamate release. This will in turn depolarize the HCs, forming the sag and/or inflection points on the response waveforms [13].
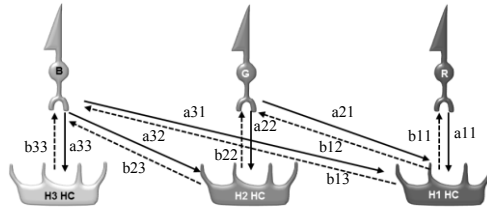
Fig. 4. Schematic diagram of the connection between cones and HCs. R, G and B are red-green- and blue- cones respectively. Solid lines indicate glutamatergic synapses, and dashed lines proton feedback synapses.

In the case of red light stimulation, green- and blue-sensitive cones will not hyperpolarize since their spectral sensitivities are nearly zero [12]. However, red cones and H1 HCs are hyperpolarized, which will induce alkalinization of cone synaptic clefts to which H1 HC dendrites are connected (dashed lines b11, b12 and b13 in Fig. 4). Alkalinization at the green-sensitive cone synaptic cleft will induce an increase in glutamate release due to increase in calcium current. This will in turn depolarize H2 HCs (line a22), which will induce acidification at the blue-sensitive cone synaptic cleft (dashed line b23) due to extrusion of proton via V-ATPase at H2 HCs. Finally, this cascade will induce a decrease in glutamate release at the synaptic cleft of blue-sensitive cones, resulting in hyperpolarization of H3 HCs (line a33). Similarly, depolarizing responses of H3 HCs to middle wavelength light is generated by alkalinization of blue cone synaptic cleft induced by hyperpolarization of H2 HCs (dashed line b23) as well as H1 HCs (dashed line b13). Under these configurations, HEPES would block changes in the synaptic cleft pH (blockade of all dashed lines); therefore, light response of H1 HCs will increase its response amplitude. Similarly, depolarizing responses of H2 and H3 HCs will be suppressed and response amplitude of the hyperpolarizing responses to short wavelengths will be increased. However, recordings from H3 HCs have revealed that, in some cases, response to red light stimulation reversed its polarity (data not shown). This result can only be interpreted by a direct feedback connection from H1 HCs to blue cones (dashed line b13), and that HEPES could only block either the feedback from H1 HCs to green cones (dashed line b12) and/or the feedback from H2 HCs to blue cones (dashed line b23). Moreover, by the observation that depolarizing responses are significantly delayed compared to the hyperpolarizing responses to short wavelength light stimulation (Figs. 2C,D & 3C,D), it can be speculated that the proton feedback pathway is delayed by about 10 – 20 ms including the synaptic delay from cones to HCs.

5   Conclusion

It has been demonstrated that HEPES suppressed depolarizing responses of H2 and H3 HCs, indicating that proton mediated feedback synapses from HCs to cones are involved in color opponent signal formation in the outer plexiform layer (OPL). These depolarizing responses were significantly delayed compared to hyperpolarizing responses to short wavelengths, as measured by differences in the delay of hyperpolarizing and depolarizing responses (§3.2). Therefore, the network connection of the OPL consists of a dynamic cascade network with delay. We are currently

developing an OPL mathematical model based on these findings to uncover the details of color signal transformation due to the proton feedback mechanism.

References

1. Hirasawa, H., Kaneko, A.: pH Changes in the Invaginating Synaptic Cleft Mediate Feedback from Horizontal Cells to Cone Photoreceptors by Modulating Ca^{2+}2+ Channels. J. Gen. Physiol. 122, 657–671 (2003)
2. Kaneko, A., Yamada, M.: S-Potentials in the Dark-Adapted Retina of the Carp. J. Physiol. 227, 261–273 (1972)
3. Tsukamoto, Y., Yamada, M., Kaneko, A.: Morphological and Physiological Studies of Rod-Driven Horizontal Cells with Special Reference to the Question of whether They Have Axons and Axon Terminals. J. Comp. Neurol. 255, 305–316 (1987)
4. Stell, W.K., Lightfoot, D.O., Wheeler, T.G., Leeper, H.F.: Goldfish Retina: Functional Polarization of Cone Horizontal Cell Dendrites and Synapses. Science 190, 989–990 (1975)
5. Perlman, I., Normann, R.A.: The Effects of GABA and Related Drugs on Horizontal Cells in the Isolated Turtle Eetina. Vis. Neurosci. 5, 469–477 (1990)
6. Barnes, S., Bui, Q.: Modulation of Calcium-Activated Chloride Current via pH-Induced Changes of Calcium Channel Properties in Cone Photoreceptors. J. Neurosci. 11, 4015–4023 (1991)
7. Tombaugh, G.C., Somjen, G.G.: Effects of Extracellular pH on Voltage-Gated Na++, K++ and Ca2+2+ Currents in Isolated rat CA1 Neurons. J. Physiol. 493, 719–732 (1996)
8. Cadetti, L., Thoreson, W.B.: Feedback Effects of Horizontal Cell Membrane Potential on Cone Calcium Currents Studied with Simultaneous Recordings. J. Neurophysiol. 95, 1992–1995 (2006)
9. Vessey, J.P., Stratis, A.K., Daniels, B.A., Da Silva, N., Jonz, M.G., Lalonde, M.R., Baldridge, W.H., Barnes, S.: Proton-Mediated Feedback Inhibition of Presynaptic Calcium Channels at the Cone Photoreceptor Synapse. J. Neurosci. 25, 4108–4117 (2005)
10. Jouhou, H., Yamamoto, K., Homma, A., Hara, M., Kaneko, A., Yamada, M.: Depolarization of Isolated Horizontal Cells of Fish Acidifies their Immediate Surrounding by Activating V-ATPase. J. Physiol. 585, 401–412 (2007)
11. Yamamoto, K., Jouhou, H., Iwasaki, M., Kaneko, A., Yamada, M.: Strongly pH-Buffered Ringer's Solution Expands the Receptive Field Size of Horizontal Cells in the Carp Retina. Zoolog. Sci. 25, 419–427 (2008)
12. Harosi, F.I.: Spectral Relations of Cone Pigments in Goldfish. J. Gen. Physiol. 68, 65–80 (1976)
13. Yang, X.L., Tauchi, M., Kaneko, A.: Convergence of Signals from Red-Sensitive and Green-Sensitive Cones onto L-Type External Horizontal Cells of the Goldfish Retina. Vision Res. 23, 371–380 (1983)
14. Yasui, S., Yamada, M.: H1 Horizontal Cells of Carp Retina have Different Postsynaptic Mechanisms to Mediate Short- versus Long-Wavelength Visual Signals. Exp. Brain Res. 74, 156–162 (1989)
15. Hare, W.A., Owen, W.G.: Effects of Bicarbonate versus HEPES Buffering on Measured Properties of Neurons in the Salamander Retina. Vis. Neurosci. 15, 263–271 (1998)
16. DeVries, S.H.: Exocytosed Protons Feedback to Suppress the Ca^{2+} Current in Mammalian Cone Photoreceptors. Neuron. 32, 1107–1117 (2001)
17. Hosoi, N., Arai, I., Tachibana, M.: Group III Metabotropic Glutamate Receptors and Exocytosed Protons Inhibit L-Type Calcium Currents in Cones but not in Rods. J. Neurosci. 25, 1062–1072 (2005)

Modeling Manifold Ways of Scene Perception

Mengyuan Zhu[1] and Bolei Zhou[2]

[1] MOE-Microsoft Key Laboratory for Intelligent Computing and Intelligent Systems,
Department of Computer Science and Engineering, Shanghai Jiao Tong University
[2] Department of Information Engineering, The Chinese University of Hong Kong
zhumengyuan@126.com, zhoubolei@gmail.com

Abstract. In this paper, under the efficient coding theory we propose a computational model to explore the *intrinsic* dimensionality of scene perception. This model is hierarchically constructed according to the information pathway of visual cortex: By pooling together the activity of local low-level feature detectors across a large regions of the visual fields, we build the population feature representation as the statistical summary of the input image. Then, a large amount of population feature representations of scene images are embedded unsupervisedly into a low-dimensional space called perceptual manifold. Further analysis on the perceptual manifold reveals the topographic properties that 1) scene images which share similar perceptual similarity stay nearby in the manifold space, and 2) dimensions of the space could describe the perceptual continuous changes in the spatial layout of scenes, representing the degree of naturalness, openness, etc. Moreover, scene classification task is implemented to validate the topographic properties of the perceptual manifold space.

Keywords: visual perception, hierarchical model, scene classification.

1 Introduction

One of the fundamental issues in computational neuroscience concerns how information is encoded and represented by the neural architecture of the brain. As for the neural computation of visual signal, from photoreceptor of retina to the unified scene perception in high-level cortex, the visual processing of input signal is hierarchically constructed. With multilayers of wiring in visual cortex, neural response gradually achieve generalization over the input signal [1].

The neurophysiologic studies [2] indicate that through the hierarchical processing of information in the visual cortex, the extremely high-dimensional input signal is gradually represented by fewer active neurons, which is believed to achieve efficient coding [3]. One of the concepts of efficient coding theory is that with the metabolic constraints the visual cortex relies on the environmental statistics to encode and represent the visual signal [4], that is, a group of neurons should encode information as compactly as possible, in order to most effectively utilize the available computing resources. Mathematically, this is expressed as to maximize the information that neural responses provide about the

visual environment. This theory has been applied to derive efficient codes for natural images and to explain a wide range of response properties of neurons in the visual cortex [4].

On the other hand, studies on the natural image statistics suggest that the natural images are usually embedded in a relatively low dimensional manifold of image space, and there is a large amount of information redundancy within the natural images [5]. According to the efficient coding theory, it could be implied that neural system would be efficiently adapted to reduce information redundancy and extract the underlying low meaningful dimensionality of natural image to form the unified scene perception, in spite of the extremely high-dimensional raw sensory input from the retina [2]. From the functional viewpoint, the hierarchical architecture of visual system could be considered as the multilayered process of nonlinear dimensionality reduction, gradually resulting in sparser and more efficient response in higher-level neurons [6].

In this paper, from the functional view of neural architecture we propose a computational model for visual scene processing. This model termed Perceptual Manifold is *data-drivenly* constructed on the natural image statistics and *hierarchically layered*: By pooling together the activity of local low-level feature detectors across large regions of the visual fields, we build the population feature representation as the statistical summary of the scene image. Then, thousands of population feature representations of scene images are extracted, and to be mapped *unsupervised* along into a low dimensional space called perceptual manifold space. Analysis of this perceptual manifold reveals that scene images which share similar perceptual similarity stay nearby in the manifold space, and the dimensions of the manifold could describe the perceptual continuous changes in the spatial layout of scenes. In addition, scene classification task is performed to validate the topographic property of the perceptual manifold space.

2   Perceptual Manifold Model

A hierarchical model called Perceptual Manifold is proposed. The architecture of the proposed model includes four cortex-like layers of computation: 1) local sparse feature encoding, 2) local higher-order feature encoding, 3)population feature encoding and 4) perceptual manifold embedding (refer to Fig.1).

Different layers of computation abstract their own representations of input signals, which accounts for the different hierarchical stages of neuronal response to visual stimulus [2]. It is assumed that through the hierarchial layers of computation, the dimensionality of image representation is gradually reduced, that is, $M < N < K$, leading to more general organization of image representation.

2.1   Local Sparse Feature Encoding

Experimental studies have shown that the receptive fields of simple cells in the primary visual cortex produce a sparse representation of input signal [4]. Efficient
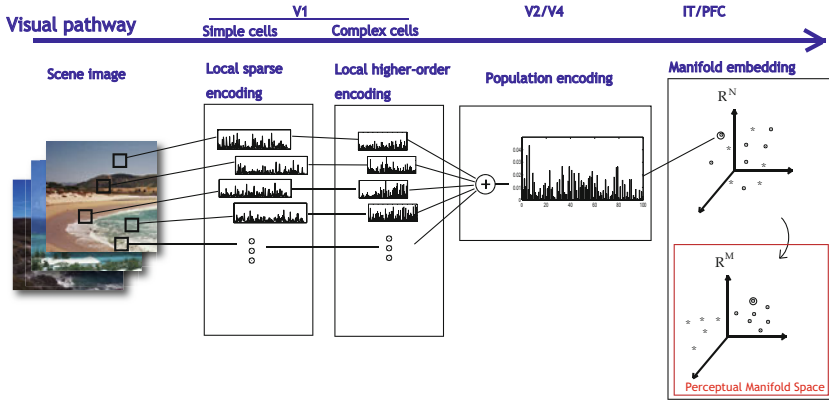
Fig. 1. Schematic diagram of the Perceptual Manifold model. There are four hierarchical layers of sensory computation to form the final perceptual space, which resemble the information pathway in visual cortex.

coding method [7] assumes that the image patch is transformed by a set of linear filters w_i to output response u_i. In matrix form,

$$\mathbf{u} = \mathbf{W}\mathbf{x} \tag{1}$$

Or equivalently in terms of generative process, $\mathbf{x}=\mathbf{A}\mathbf{u}=\mathbf{W}^{-1}\mathbf{u}$. Then, the filter response u_i are assumed to be statistically independent,

$$p(\mathbf{u}) = \prod_i p(u_i) \tag{2}$$

where $p(u_i) \propto \exp(-|u_i|)$. Let $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, ...\mathbf{w}_K]$ be the learned filter functions, and K is the number of basis functions, so that the $\dim(\mathbf{u})=K$. Fig. 2 shows a subset of the filter functions $\mathbf{w}$. These filter functions resemble the receptive field properties of simple cells, i.e., they are spatially localized, oriented and band-pass in different spatial frequency bands. A vectorized image patch $\mathbf{x}$ can be decomposed into those statistically independent bases, in which only a small portion of bases are activated at one time. They are used as the first layer of local feature extraction in our framework, so that representation of local image patches in first layer is $\mathbf{u}$. This layer of computation resembles the simple cells in V1 [8].

2.2   Local Higher-Order Feature Encoding

Higher-level visual neurons encode statistical variations that characterize local image regions, these results provide a functional explanation for nonlinear effects in complex cells [9]. Thus the coefficients of local basis $\mathbf{A}$ are further assumed to follow a generalized Gaussian distribution,

$$p(u) = \mathcal{N}(0, \lambda, q) = z\exp(-|\frac{u}{\lambda}|^q), \tag{3}$$
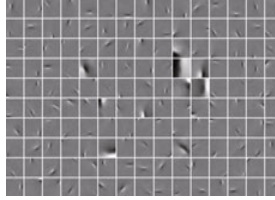
Fig. 2. A subset of filter basis $\mathbf{W}$ trained from natural image patches

where $z = q/(2\lambda\Gamma[1/q])$ is the normalizing constant, and mostly $\lambda=1$. In [10], the variance λ value is assumed to be generated by variance basis as follows:

$$\log\boldsymbol{\lambda} = \mathbf{B}\mathbf{v}, \tag{4}$$

where $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, ...\mathbf{b}_N]$ are variance basis functions trained from thousands of natural image patches[1], N is the number of variance basis functions and $\mathbf{v}$ is the higher-order representation of local image patches, so that the $\dim(\mathbf{v})=N$, where $N<K$. The transformation from sparse representation $\mathbf{u}$ to the higher-order representation $\mathbf{v}$ is determined by maximizing the posterior distribution for a given $\mathbf{u}$,

$$\hat{\mathbf{v}} = \operatorname{argmax} p(\mathbf{u}|\mathbf{B}, \mathbf{v})p(\mathbf{v}). \tag{5}$$

where $p(\mathbf{v})=\prod_j p(v_j)$ and $p(v_j) \propto \exp(-|v_j|)$. In the simulation, $\hat{\mathbf{v}}$ is derived by gradient ascent [10]. For simplicity, here the nonlinear transformation of $\mathbf{u}$ to $\mathbf{v}$ is denoted as a general function f, where $\mathbf{v}=f(\mathbf{u})$. This is second layer of computation in our framework, and through it $\mathbf{v} = [v_1, v_2, ..., v_N]^\top$ becomes the representation of local image patches. This layer of computation resembles the complex cells in V1 [9].

2.3 Population Feature Encoding

The neurophysiologic study [2] suggests that on the population level in extrastriate visual areas II(V2) and IV(V4), a normalized pooling mechanism might be used to extract the global response of the stimulus. Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n, ...]$ denote the sample matrix, where $\mathbf{x}_n$ is the vectorized image patch sampled from one scene image. After the first two layers of local encoding: $\mathbf{u}_n = \mathbf{W}\mathbf{x}_n$ and $\mathbf{v}_n = f(\mathbf{u}_n)$, the population feature component for the ith feature of $\mathbf{v}_n$ is,

$$p_i = \frac{\sum_n ([\mathbf{v}_n]_i)^2}{\sum_{i=1}^N \sum_n ([\mathbf{v}_n]_i)^2} \tag{6}$$

where $[\mathbf{v}_n]_i$ indicates the ith element of the vector $\mathbf{v}_n$. Thus, $\mathbf{p} = [p_1, p_2, ..., p_N]^\top$ indicates the normalized population feature response of scene image, which accounts for the holistic representation of scene image in the third layer of computation. This layer of computation resembles the population coding in V2/V4.

---

[1] Refer to [10] for visualization of the variance basis.

2.4   Perceptual Space Embedding

To explore the intrinsic dimensionality of scene perception, the Local Linear Embedding [11] is applied as the method of nonlinear dimensionality reduction to a large amount of the population feature responses of different scenes:

First step: compute the weight ω_{ij} that best linearly reconstructs $\mathbf{p}_i$ from its neighbor $\mathbf{p}_j$, minimizing:

$$\varepsilon(\omega) = \sum_i |\mathbf{p}_i - \sum_j \omega_{ij}\mathbf{p}_j|^2 \tag{7}$$

Second step: compute the low-dimensional embedding vectors $\mathbf{q}_i$ best reconstructed by ω_{ij}, minimizing:

$$\phi(\mathbf{q}) = \sum_i |\mathbf{q}_i - \sum_j \omega_{ij}\mathbf{q}_j|^2 \tag{8}$$

The resulting embedding space $\mathbb{R}^M$ is called perceptual manifold space, as the final layer of computation in our architecture. And $\mathbf{q} = [q_1, q_2, ..., q_M]^\top$ is the representation of scene perception for a specific image, in which $\dim(\mathbf{q})=M$ and $M < N < K$. This layer of computation is believed to exist in the inferotemporal cortex(IT) or the prefrontal cortex(PFC), which involve in forming the perception of objects and scenes [12].

The implementation and analysis of the perceptual manifold space are presented in the following experiment section.

3   Experiments

For the training of image basis $\mathbf{A}$(or $\mathbf{W}$) and variance basis $\mathbf{B}$, 150000 20×20 gray image patches from a standard set of ten $512{\times}512$ natural images [13] are extracted. The number of 20×20 filter basis $\mathbf{W}$ is 400, equivalently K=400, and the number of variance basis function $\mathbf{B}$ is limited to 100, equivalently N=100. For manifold embedding layer, the dataset of scene images used here comes from [14], which contains 3890 images from 13 semantic categories of natural scenes, like coast and forest, etc. All 3890 images are normalized to $128{\times}128$ pixels before layers of computation. The dimensionality of manifold space M is tuned empirically as 15, so that the manifold embedding is $\mathbb{R}^{100} \to \mathbb{R}^{15}$. Thus, through the whole process of multilayered computations, the representative space changes as $\mathbb{R}^{400} \to \mathbb{R}^{100} \to \mathbb{R}^{15}$.

In the following part, the topographic properties of the perceptual space are analyzed at first. Then to validate those properties of perceptual space, scene classification task is performed in this perceptual space.

3.1   Perceptual Space Analysis

For visualization of the perceptual space, data points of scene images from four scene categories are described by the first three principal component coordinates
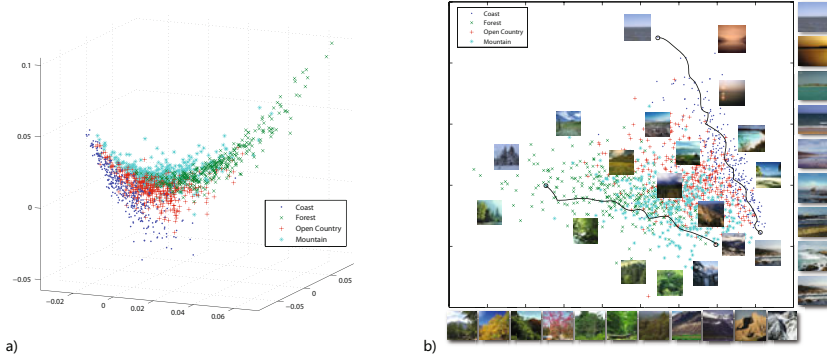
Fig. 3. Data points of scene images from four categories are visualized by the first three coordinates and first two coordinates of perceptual space. **a)** Clustering and nonlinear geometric property of the data points in the 2D perceptual space. **b)** Representative scenes are shown next to the corresponding data points in different parts of the 3D perceptual space. The bottom and right sets of images correspond to points along the two pathes(linked by solid line), illustrating particular perceptual changes in scene images.

(Fig. 3a) and first two principal component coordinates of perceptual space (Fig. 3b). In Fig. 3a, we can see that there are clustering and nonlinear geometric properties among the pool of data points. In Fig. 3b, representative scenes are shown next to the corresponding data points in different regions of the perceptual space. The bottom and right sets of images correspond to points along the two pathes(linked by solid line), illustrating particular degree of perceptual changes within the scene images.

As we can see, the topographic properties of perceptual space are related to the perceptual dimensions(degree of naturalness, openness, expansion, etc) supervised trained in [15], which represent the dominant spatial dimensions of a scene. Our Perceptual Manifold model is layered in a bottom-up way to find the *intrinsic* dimensionality of scene perception. Our finding supports the viewpoint that the shape of a scene could be described by a few perceptual dimensions [16]. Moreover, the topographic properties give further implication that human visual system might be adapted to both *extract* and *integrate* the lower perceptual dimensions to form the holistic scene perception. After that, scene classification task is performed to validate the topographic properties of perceptual manifold space.

3.2   Scene Classification

Scene classification task is to classify each image from testing set into one category of scenes. The dataset contains 13 categories of scenes, 100 images from each scene class are as training set, and the rest are as testing set. Both testing

set and training set of images have been embedded in perceptual manifold space before the classification, so that all images are represented as 15 dimensional feature vectors. A 13-way linear SVM classifier is trained on the training set, then it is applied to classify images from testing set.

The average accuracy of classification for our method is 68.9%. The average accuracy for baseline methods LDA[14] is 64.0%. Even though the Perceptual Manifold model is not designed specifically for the scene classification task, our model achieves good performance. Scene classification task well validates the topographic properties of the perceptual manifold space. And it further reveals that there is neural correlation between the perceptual space and semantic space in human cognitive process [16].

4   Discussion

4.1   On the Dimensionality of Perceptual Manifold Space

The choice of reduced dimensionality M for manifold space is theoretically and experimentally important. First, the local linear embedding [11] itself relies on the amount of observation samples and the setting of reduced dimensionality to search for intrinsically low-dimensional structures embedded nonlinearly in high-dimensional observations. And the theoretic analysis of this point could be found in the studies on manifold learning [17], which goes beyond the scope of this paper. Second, the topographical properties of perceptual manifold are influenced by the dimensionality M value, here we illustrate that by the correlation between dimensionality M value and performance of scene classification, as shown in Fig. 4. From that we can see, the dimensionality to engender the topographic properties of perceptual manifold is rather low. For the limitation of paper length, more theoretic analysis would be included in our further work.
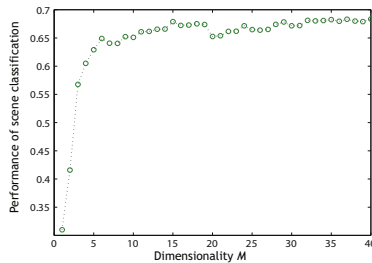
Fig. 4. The correlation between dimensionality M and scene classification performance

5   Conclusion

A novel hierarchical model of scene perception termed Perceptual Manifold is introduced in this paper. Through the cortex-like layers of computation, dimensionality of input visual signals is gradually reduced, and it finally leads to the

formation of perceptual manifold space. In this perceptual manifold space, there exist topographic properties that **1)** data points of perceptual similarly scene images stay nearby in this perceptual manifold space and **2)** dimensions of the perceptual space could describe the meaningful continuous changes in the spatial layout of scene images. Scene classification task is performed to validate the topographic properties of the perceptual manifold space.

References

1. Nassi, J.J., Callaway, E.M.: Parallel processing strategies of the primate visual system. Nat. Rev. Neurosci. 10(5), 360–372 (2009)
2. Weliky, M., Fiser, J., Hunt, R.H., Wagner, D.N.: Coding of natural scenes in primary visual cortex. Neuron (2003)
3. Olshausen, B.A., Field, D.J.: Sparse coding of sensory inputs. Current Opinion in Neurobiology 14(4), 481–487 (2004)
4. Simoncelli, E.P., Olshausen, B.: Natural image statistics and neural representation. Annual Review of Neuroscience (2001)
5. Srivastava, A., Lee, A.B., Simoncelli, E.P., Zhu, S.-c.: On advances in statistical modeling of natural images. Journal of Mathematical Imaging and Vision (2003)
6. Seung, S.H., Lee, D.D.: Cognition: The manifold ways of perception. Science (2000)
7. Bell, A.J., Sejnowski, T.J.: The "independent components" of natural scenes are edge filters. Vision Res. (1997)
8. van Hateren, J., van der Schaaf, A.: Independent component filters of natural images compared with simple cells in primary visual cortex. Proc. Royal Soc. Lond. B 265, 359–366 (1998)
9. Karklin, Y., Lewicki, M.S.: Emergence of complex cell properties by learning to generalize in natural scenes. Nature 457, 83–86 (2009)
10. Karklin, Y., Lewicki, M.S.: A hierarchical bayesian model for learning nonlinear statistical regularities in nonstationary natural signals. Neural Comp. 17(2), 397–423 (2005)
11. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science (2000)
12. Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., Poggio, T.: Robust object recognition with cortex-like mechanisms. IEEE Trans. Pattern Anal. Mach. Intell. 29(3), 411–426 (2007)
13. Olshausen, B.A., Field, D.J.: Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature 381(6583), 607–609 (1996)
14. Li, F.F., Perona, P.:
15. Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. International Journal of Computer Vision (2001)
16. Oliva, A.: Gist of the scene. In: Itti, L., Rees, G., Tsotsos, J.K. (eds.) The Encyclopedia of Neurobiology of Attention, pp. 251–256. Elsevier, San Diego (2005)
17. Saul, L.K., Weinberger, K.Q., Ham, J.H., Sha, F., Lee, D.D.: Spectral methods for dimensionality reduction. Semisupervised Learning. MIT Press, Cambridge (2006)

Utilization of a Virtual Patient Model
to Enable Tailored Therapy for Depressed Patients

Fiemke Both and Mark Hoogendoorn

VU University Amsterdam, Department of Artificial Intelligence
De Boelelaan 1081, 1081 HV Amsterdam, The Netherlands
{f.both,m.hoogendoorn}@vu.nl

Abstract. Major depression is a prominent mental disorder that has significant impact upon the patient suffering from the depression as well as on the society as a whole. Currently, therapies are offered via the Internet in the form of self-help modules, and they have shown to be as effective as face-to-face counseling. In order to take automated therapies a step further, models which describe the development of the internal states associated with depression can be of great help to give dedicated advice and feedback to the patient e.g. by means of making predictions using the model. In this paper, an existing computational model for states related to depression (e.g. mood) is taken as a basis in combination with models that express the influence of various therapies upon these states. These models are utilized to give dedicated feedback to the patient, tailor the parameters towards the observed patient behavior, and give an appropriate advice regarding the therapy to be followed.

Keywords: Virtual patient, depression, support agent, tailored advice.

1 Introduction

Major depression is currently the fourth disorder worldwide in terms of disease burden, and is expected to be the disorder with the highest disease burden in high-income countries by the year 2030 according to a prediction of the World Health Organization (WHO). Within mental health care, a new generation of therapies for treatment of depression has emerged, in which patients can use Internet-based self-help therapies. This takes away the long waiting times for psychological treatments and also removes the barrier of going to a doctor to seek counseling. A wide variety of therapies are available in the form of self-help modules, including activity scheduling (also called behavioral activation, see e.g. [14]), cognitive behavior therapy (see e.g. [3]), and problem solving therapy (see e.g. [10]). A growing number of randomized trials have been performed that show that such forms of treatment are as effective as face-to-face treatment (see e.g. [15]; [1]). However, these treatments currently do not provide very personalized or tailored support to the patient, which could potentially lead to an even better treatment of the depression.

In order to provide such personal advice and support, the supporting system (for instance in the form of a personal support agent) should be able to build up a picture of the current and potential future development of the patient, and the influence of the

therapy upon this development. Based upon this picture, the system can determine how the patient is progressing and is expected to progress further, provide feedback on this, and also select the most appropriate therapy for the patient. In previous work [5], a computational model for the cognitive states associated with depression has been developed based upon literature available in clinical psychology, including elements such as *mood*, *thoughts*, and *appraisal* and their interrelationship. This model can be used to predict the development of these states of the patient over time. In extensions of the model (see [6];[7]), the influence of various therapies upon these states of the patient have been included, which enables predictions on the effectiveness of the therapy for the patient.

In this paper, an approach is presented which utilizes such models to provide dedicates feedback to the patient and give advice on the appropriate therapy to follow. This process is composed of three subparts: (1) deriving the therapy with the highest probability of success and providing this as advice to the patient, (2) once a therapy has been selected, the predicted trends can be compared with the observed trends of the patient to provide feedback, but also to trigger a process which evaluates whether there is potentially a more effective therapy than the current one, and (3) tuning the parameters of the predictive models towards the observations with respect to the patient in case large deviations are found to improve the accuracy of future predictions.

This paper is organized as follows. Section 2 briefly describes the underlying models. In Section 3 the algorithms that describe how such models can be utilized are described, whereas Section 4 presents simulations results for a dedicated scenario. Finally, Section 5 is a discussion.

2   Therapeutic Models

In [6] and [7], therapies are described in terms of a computational mood model [5]. In Figure 1, the mood model is shown in gray. This model uses situations in the world (*world events*) and personal characteristics such as *coping skills*, *vulnerability* and *prospected mood level* to describe the state of a person. In the figure, the black arrows and grey circles describe the therapeutic influence of the Cognitive Behavior Therapy (CBT) module with three effects on the mood model: *intervention*, *reflection* and a therapy-dependent effect, *appraisal* in case of CBT. The first two hold for all interventions.

- *Intervention* effect: the positive effect on the thoughts of a person when this person is following a form of therapy.
- *Reflection:* the learning effect of a therapy by increasing coping skills. The concepts involved in reflection differ among the therapies.
- Therapy-dependent effect: each intervention has its own view on how to treat depression. CBT focuses on changing the appraisal of a situation, the other two modeled interventions focus on introducing more pleasant (activity scheduling) or physical (exercise therapy) activities into a person's daily life. These effects are described in more detail below.

The idea behind CBT is that there is a relation between how a situation is appraised (*appraisal*) and the mood level. During the therapy, one learns that emotions are triggered by thoughts about a situation (*appraisal*) rather than by the situation itself.
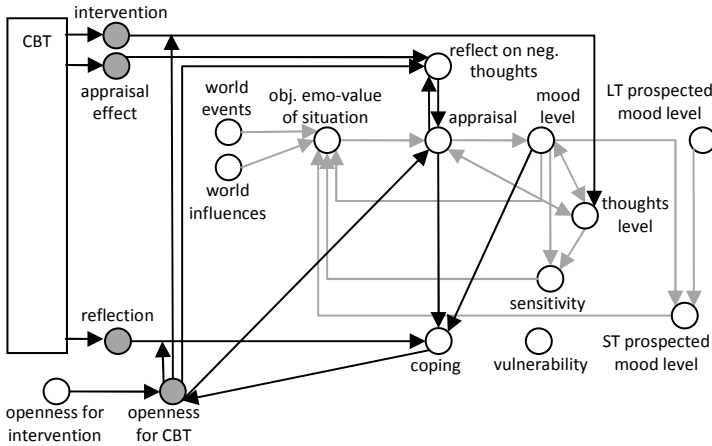
Fig. 1. Computational model for Cognitive Behavior Therapy (CBT). The model for dynamics of mood and depression is shown in gray. The additions for the CBT model are shown in black.

These negative thoughts can be identified and changed into thoughts that are more positive. This is modeled by an *appraisal effect* from the CBT module influencing the element *reflect on negative thoughts,* which in turn influences *appraisal*. Openness for CBT determines to what extent these concepts are influenced. A person that is very open to a specific therapy will put more effort into the therapy and will learn quicker.

The other therapies modeled in previous work are Activity Scheduling and Exercise Therapy. During Activity scheduling (AS), the patient learns the relationship between the selection of a relatively positive activity and the level of mood (i.e., when you do fun things, you will start to feel better, based on the reinforcement theory in [14]). In order to learn this relationship, the therapy imposes the selection of positive situations. Exercise Therapy (ET) is based on the idea that physical exercise may improve mood ([4], [8]). A number of concepts have been added to the model, such as *physical state* and *physical norm*. The physical state influences the mood level positively and learning the relation between these concepts increases the coping skills. For more details on the mood model and the models of the different therapies, see ([5], [6], [7])

3   Support System Utilizing Therapeutic Models

In order to utilize the models described in Section 2, this Section introduces a three stepped approach. The first step comprises of the utilization of the models to advise the patient which therapy to follow. Once a therapy has been selected, comparisons

can be made between the predicted developments of the patient in terms of the mood level and the actual development of the patient based upon measurements. This can either be used as a basis for feedback (e.g. "*you're developing much faster than expected, congratulations!*") or to trigger a new process to see whether a switch in therapy could result in a more speedy recovery. In case the predictions of the model show to be inaccurate, the third step is to perform parameter adaptation to improve the predicative power of the models. Each of these elements is treated in more detail below.

3.1   Selecting a Suitable Therapy

When a patient wants to start with a treatment, the first step to undertake is to determine what type of therapy the patient should follow. In the approach presented in this paper, the models which describe the development of internal states and the influence of therapy upon this development are utilized to advice the most appropriate therapy. Simulations are performed to see how the mood of the patient will develop, given a certain therapy which has been selected. In these models, a number of parameters are present that depend on the characteristics of the patient (as described in Section 2). Specifically, the following parameters are assumed to be set in accordance with such patient dependent characteristics:

1. Initial level of mood
2. Coping skills level
3. Vulnerability level
4. Openness for the type of therapy

Hereby, the first element (initial level of mood) can be measured directly ("*what is your mood on a scale from 1-10?*"), whereas the second and third parameter (i.e. coping and vulnerability) follow from a set of dedicated questions that are part of an initial questionnaire the patient has to fill in. The parameter openness for therapy depends on the prior experience of the patient (has the therapy been followed before, and if so, was it successful), and the general characteristics (e.g. does the patient like to run to determine whether the patient is open towards exercise therapy). Once precise values are derived for each of these elements, predictive simulations for each of the different therapies can be run. The criterion for advising the best therapy is simply the therapy in which the mood level will be 6 or higher during three consecutive days (i.e. sufficiently high again) within the shortest time span. If none of the therapies meets this criterion, the therapy with the highest mood level at the end of the therapy is advised.

3.2   Comparing Predictions and Actual Observations

Once the patient has decided to follow a certain therapeutic module, the support agent can monitor the effectiveness of the therapy by comparing the predictions of the models with the actual observations around the patient. Such a comparison is not a trivial task. A precise comparison between the predicted and observed values is often difficult to do, as one more wants to see whether the trends are similar, not whether the precise numbers are the same. Therefore, first trends are expressed within the

development of the states of the patient which are applicable to both the predictions using the model and the actual observations of the patient. Thereafter, these trends are compared and conclusions are drawn, possibly resulting in actions being undertaken.

Identification of Trends. Making a comparison between the data which has been collected from the patient (e.g. due to the mood ratings the patient has performed) and the predictions of the model is difficult. The model tends to make predictions that are relatively smooth and give a more general indication of the trends of the patient, whereas the observations around the patient are fluctuating a lot more, for instance during the start of the day the mood is generally rated a lot lower compared to the middle of the day. As a consequence, a comparison based upon the general trends of the predictions of the model and the general trends in the observed data of the patient is much more useful. The following trends are hereby distinguished:

- *Increasing during a period x:* The general trend is that a particular aspect of the therapy or state of the patient is increasing during a certain time period x.
- *Decreasing during a period x:* The general trend is that a particular aspect of the therapy or state of the patient is decreasing during a certain time period x.
- *Stable (fluctuations within certain boundaries) during a period x:* The general trend for a particular aspect of the therapy or the state of the patient is stable during a period x.
- *Average over a period x is above a threshold th:* The average value for a particular aspect of the therapy or the state of the patient is above a certain threshold value th during a certain time period x.
- *Average over a period x is below a threshold th:* The average value for a particular aspect of the therapy or the state of the patient is below a certain threshold value th during a certain time period x.

All of these trends are expressed in more detail below.

Variable v increasing during a period x.
In order to express increasing trend with respect to some measurements around the patient is not a trivial matter. Certain outliers might occur in the data that need to be filtered out, and when looking at individual measurements such outliers can be quite difficult to detect. For instance, sometimes a clear increasing trend can be seen, but rare outliers prohibit a strict property with respect to an increasing measurement from being satisfied. Of course, many different techniques can be applied to detect the increasing trends, e.g. the fitting of a linear curve through the data making use of the method of least squares (e.g. [13]). In this case, another approach has been used which averages the measurements over the days and detects whether these averages are monotonically increasing. Due to the fact that the predictions as given by the models provide quite a lot of data per day, this approach is computationally more efficient, and is also closer to the current approaches used in clinical psychology. The algorithm to derive whether this criterion is indeed fulfilled is expressed on the next page. Note that hereby a start and end time are assumed. The duration between these time points should be equal to x (the duration expressed in the property).

In the algorithm, a loop is present in which the average is taken for a window from the current time point till the current time point plus a duration d (the window size).

In case this average is strictly higher than the average in the previous window, the property can still be satisfied, and the loop continues by setting the current time to a new value. In case the loop is completely passed the property succeeds. If a case occurs whereby the average is not higher than the previous average, the property fails and the cycle ends.

Algorithm 1. *Increasing trend from start time t_{start} to end time t_{end}*

```
t_current = t_start
previous_average = 0;
while (t_current < t_end) {
        total = 0;
        for (int i=0; i < d; i++){
                total = total + v(t_current + i);
        }
        current_average = total / d;
        if (current_average ≤ previous_average) {
                return false;
        }else{
                previous_average = current_average;
                t_current = t_current + d;
        }
}
return true;
```

Variable v decreasing during a period x

For the decreasing trend, a similar approach can be taken as described for the increasing trend.

Variable v stable (fluctuations within certain boundaries) during a period x

Furthermore, a stable trend is also expressing, which indicates that the variable v fluctuates within certain boundaries.

Algorithm 2. *Stable trend from start time t_{start} to end time t_{end}*

```
t_current = t_start
average = 0;
total = 0;
timesteps = 0;
for (int t=t_start; t < t_end; t++){
        timesteps++;
        total = total + v(t)
}
average = total / timesteps;
for (int t=t_start; t < t_end; t++){
        if (v(t) > (1+b) * average || v(t) < (1-b) * average){
                return false;
        }
}
return true;
```

Hereby, once a value exists which deviates more than *b* from the average value during the entire period, the algorithm returns false.

Average of variable v over a period x is above/below a threshold th.
For the sake of brevity, the algorithm underlying this definition is not shown, but for a calculation of the average value algorithm 2 can be followed, and a simple check can be performed to see whether this average is above or below the threshold value th.

Comparison of Trends. Once the trends for both the actual patient states and the predicted patient states are known, a comparison can be made to see how these trends relate to each other. In this case, a comparison is based upon the general level within the patient (the patient is doing well when the average level is above six) as well as the trend in the development of the measurement (i.e. stable, increasing or decreasing). In Table 1 a categorization is given of the comparison between the predicted and observed trend regarding the development of the patient.

Table 1. Comparison between trends

Observed trend	Predicted trend	good			bad		
		increasing	stable	decreasing	increasing	stable	decreasing
good	increasing	o	+	++	++	++	++
	stable	-	o	+	+	+	++
	decreasing	--	-	o	o	o	o
bad	increasing	o	o	o	o	+	++
	stable	--	-	-	-	o	+
	decreasing	--	--	--	--	-	o

In the table, a '--' expresses a significant worse development of the patient's state compared with the predictions. A '-' indicates a somewhat worse development, and a 'o' represents a development comparable with the prediction. '++' is a far more positive development, whereas '+' is a somewhat more positive development. In all cases, except when the patient is performing significantly worse, feedback is given to the patient how he/she is doing compared with the predictions. All of these messages are phrased positively and meant to stimulate the patient as much as possible. Examples of such messages include: "*you're doing much better than other people in your situation, keep up the good work!*" for the case of '++' and "*you're progression is a little bit less than expected, try as best as you can to get the most out of the therapy and enable a rapid recovery!* ". If the performance is significantly worse than expected, a process is started to seek for an alternative therapy which might be more suitable for the patient. The first step in this process is to tune the parameters of the models used to predict the patient development towards the observed data of the patient.

3.3   Tuning the Model towards Observed Patient Behavior

Once it has been shown that the trend in the patient state is much worse than expected, apparently the predictions of the models were not sufficiently accurate. Hence, the parameters of the models should be adapted such that the model describes

the behavior of the patient more precise. Thereafter, the models can again be utilized to select the most appropriate therapy (cf. Section 3.1) with these newly gained insights about the parameters of the patient. In the parameter adaptation process, two parameters are adapted, namely coping and vulnerability. For the openness of therapy the old value is assumed (as it has developed as a result of the model), and for the initial level of mood the actual input of the patient is used. Algorithm 3 expresses the algorithm for the parameter adaptation process.

```
Algorithm 3. Parameter adaptation
current_best_value_coping = low;
current_best_value_vulnerability = low;
current_best_mse = 1; // maximum value

for all settings for coping
        for all setting for vulnerability
                    current_mse = mse(current_value_coping, current_value_vulnerability,
                                            current_therapy);
                    if (mse(current_mse < current_best_mse) {
                            current_best_mse = current_mse;
                            current_best_value_coping = current_value_coping;
                            current_best_value_vulnerability = current_value_vulnerability;
                        end
        end
end
```

The mean squared error is used as a measure of the fitness of the parameters. Furthermore, a limited set of parameter values is assumed to avoid a too high computational load. Once the ideal parameters have been selected, the different alternative therapies can again be compared, following the approach described in Section 3.1. Hereby it is assumed that the previous values of all the states are taken as an initial value when starting the run the predictions.

4   Simulation Results

In this Section, the overall approach is evaluated by means of simulation runs with a typical patient. The figures below show the different steps of one simulation of a person with low coping skills, high vulnerability, a low initial mood level and medium openness to therapy. The first step of the support system is to select a suitable therapy. Figure 3 shows the predicted mood level given the parameters above for the therapies AS, CBT and ET. The advice given to the person is to follow exercise therapy, because a sufficiently high mood level is reached within the shortest time span (52 days for ET, 59 days for AS and 53 days for CBT). In the figure, the solid black line indicates the threshold for the patient state in order to be considered sufficiently high.

Following step 2, the situation is evaluated after week 7 of the exercise therapy. In Figure 4a, the actual patient state, based on the reported mood level, is shown as a solid blue line. The predicted state of the patient is shown as a red striped line. The trends are depicted as a circle for the virtual and an asterisk for the actual trend, where a value of 1 means increasing, 0.5 means stable and 0 means decreasing. The trend of
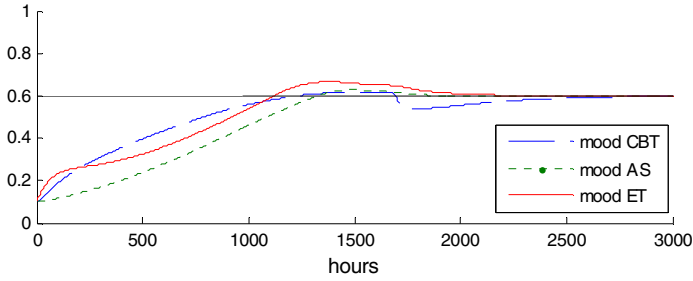
Fig. 3. Predicted mood levels for the three therapies

the prediction (red circle) is good and increasing whereas the actual trend is bad and decreasing (blue asterisk). According to Table 1, this situation is undesirable and the parameter adaptation process is started as step 3 of the support system.
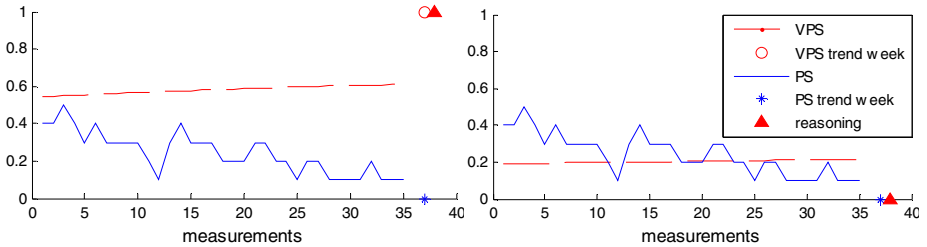
Fig. 4a, b. Virtual vs. actual patient state in week 7 of ET before (left) and after (right) parameter adaptation

Running algorithm 3 results in a new set of parameters describing the current state of the patient more precisely. The predictions based on the new parameters are shown in Figure 4b. The best fitting coping skills level is very low, in combination with a very high level of vulnerability. The mean squared errors for the coping levels very low, low, medium and high in combination with opposite vulnerability levels are respectively: 0.02, 0.13, 0.19 and 0.30. Combining corresponding levels for coping and vulnerability lead to even greater mean squared errors.

Since the patient is not doing very well (the patient state trend is bad and decreasing), the support system can now start with step 1 again: selecting a suitable therapy. All therapies are simulated again to see if the patient is better of switching to a different therapy. It is assumed that switching to a different therapy is experienced as positive, due to the personal advice that is given. Therefore, the simulations of the other therapies than the current start with a mood level one point higher than the last reported mood level. Figure 5 depicts the mood levels of the three therapies given the current state of the patient and the newly determined parameters. The support system advises to switch to CBT, because it is expected that CBT leads to the highest mood level at the end of the therapy.
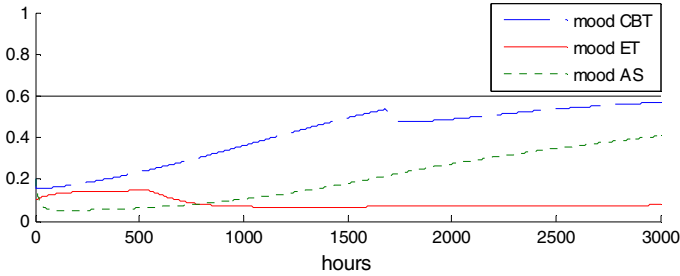
Fig. 5. Predicted mood levels for the three therapies

5   Discussion

In this paper an approach has been presented to provide feedback and give personalized advice to people suffering from a depression. In order to come to such an approach, cognitive models which represent the states associated with depression are utilized, in combination with extensions expressing the influence of dedicated therapies upon the mental states of a human. The presented approach consists of three main elements: (1) the utilization of the models to make predictions of effectiveness of therapy and give advice on the therapy with the fasted expected recovery, (2) the comparison of the predictions using the models with the actual observed behavior to provide feedback to the patient and potentially trigger a process to advise a therapeutic change, and (3) the tailoring of the parameters of the models towards the behavior of the patient in case this is necessary to guarantee the accuracy of the predictions. The overall approach has been evaluated by means of simulations runs, and shows that the approach indeed works as expected. Next steps include the validation of the models themselves with empirical data, after which the presented approach will be deployed as part of a system to be tested with actual patients.

Within the literature, several computational models incorporating emotions have been presented. For instance, [2] presents an example model which involves emotions and the influence thereof upon the behavior of an agent. Other examples of models include [9] in which agents are programmed that involve emotions in their deliberation process, and many more exist. In this paper, the utilization of these models is taken one step further, namely to give advice based upon predictions using these models. With respect to the identification of trends in the development of the patient (which is a necessity to make a comparison possible) as a first step a method has been selected which is close to the current approach followed by therapists, but more advanced computational methods could be utilized. Furthermore, for the parameter adaptation an exhaustive search approach based upon a limited set of allowed parameter values has been selected to guarantee the robustness of the model as well as to avoid overfitting. Of course, the disadvantage is that large discrepancies can still be present. Therefore, for future work, more advanced parameter adaptation techniques will be used, for instance using more mathematical-based approaches (see e.g. [12]) or Artificial Intelligence learning techniques such as Genetic Algorithms (see e.g. [11]).

References

1. Andersson, G., Bergstrom, J., Hollandare, F., Carlbring, P., Kaldo, V., Ekselius, L.: Internet-based self-help for depression: randomised controlled trial. The British J. of Psy. 187 (November 2005)
2. Bates, J., Loyall, A.B., Reilly, W.S.: Integrating Reactivity, Goals, and Emotion in a Broad Agent. In: Proceedings of the 14th Annual Conference of the Cognitive Science Society (1992)
3. Beck, A.T.: Depression: Causes and Treatment. University of Pennsylvania Press (1972)
4. Biddle, S.J.H.: Emotion, mood and physical activity. In: Physical Activity and Psychological Well-being. Routledge, London (2000)
5. Both, F., Hoogendoorn, M., Klein, M.A., Treur, J.: Formalizing Dynamics of Mood and Depression. In: Ghallab, M. (ed.) Proceedings of the 18th European Conference on Artificial Intelligence, ECAI 2008, pp. 266–270. IOS Press (2008)
6. Both, F., Hoogendoorn, M., Klein, M.C.A., Treur, J.: Computational Modeling and Analysis of Therapeutical Interventions for Depression. In: Yao, Y., Sun, R., Poggio, T., Liu, J., Zhong, N., Huang, J. (eds.) BI 2010. LNCS (LNAI), vol. 6334, pp. 274–287. Springer, Heidelberg (2010)
7. Both, F., Hoogendoorn, M., Klein, M.C.A., Treur, J.: Computational Modeling and Analysis of the Role of Physical Activity in Mood Regulation and Depression. In: Wong, K.W., Mendis, B.S.U., Bouzerdoum, A. (eds.) ICONIP 2010, Part I. LNCS (LNAI), vol. 6443, pp. 270–281. Springer, Heidelberg (2010)
8. Cotman, C.W., Berchtold, N.C.: Exercise: a behavioral intervention to enhance brain health and plasticity. Trends Neurosci. 25, 295–301 (2002)
9. Dastani, M., Meyer, J.J.C.: Programming Agent with Emotions. In: Proceedings of the European Conference on Artificial Intelligence, ECAI 2006 (2006)
10. D'Zurilla, T.J., Nezu, A.: Social problem solving in adults. In: Kendall, P.C. (ed.) Advances in Cognitive-behavioral Research and Therapy, vol. 1, pp. 202–274. Academic Press, New York (1982)
11. Kirkpatrick, S., Gelatt, C.D., Vechhi, M.P.: Optimization by simulated annealing. Science 220(4598), 671–680 (1983)
12. Koch, K.R.: Parameter Estimation and Hypothesis Testing in Linear Models. Springer, Heidelberg (1999)
13. Kreyszig, E.: Advanced Engineering Mathematics, 8th edn. John Wiley and Sons (1999)
14. Lewinsohn, P.M., Youngren, M.A., Grosscup, S.J.: Reinforcement and depression. In: Dupue, R.A. (ed.) The Psychobiology of Depressive Disorders: Implications for the Effects of Stress, pp. 291–316. Academic Press, New York (1979)
15. Spek, V.R.M., Nyklicek, I., Smits, N., Cuijpers, P., Riper, H., Keyzer, J.J., Pop, V.J.M.: Internet-based cognitive behavioural therapy for subthreshold depression in people over 50 years old: A randomized controlled clinical trial. Psychological Medicine 37(12), 1797–1806 (2007)

Learning Based Visibility Measuring with Images

Xu-Cheng Yin[1], Tian-Tian He[2], Hong-Wei Hao[1],
Xi Xu[1], Xiao-Zhong Cao[3], and Qing Li[2]

[1] Department of Computer Science, University of Science and Technology
Beijing, Beijing 100083, China
[2] Department of Automation, University of Science and Technology
Beijing, Beijing 100083, China
[3] Meteorological Observation Center, China Meteorological
Administration, Beijing 100081, China
{Xuchengyin,hhw}@ustb.edu.cn, zmhetian@163.com,
xuxihasan@yahoo.com.cn, caoxzh@126.com, liqing@ies.ustb.edu.cn

Abstract. Visibility is one of the major items of meteorological observation. Its accuracy is very important to air, sea and highways and transport. A method of visibility calculation based on image analysis and learning is introduced in this paper. First, visibility image is effectively represented by contrast based vision features. Then, a Supported Vector Regression (SVR) based learning system is constructed between image features and the target visibility. Consequently, visibility can be measured directly from a single inputting image with this learning system. The method makes use of the existing video cameras to calculate visibility in real time. Specific experiments show that this method has the characteristic of low cost, fast calculation, and convenience. Moreover, our proposed technology can be used anywhere to measure visibility.

Keywords: Visibility measuring, Learning, Image contrast.

1 Introduction

Visibility is an important item of meteorological observation. It is used in military, environment and climate change and so on. Its accuracy is vital to air, sea and highways and transport. Besides, it can provide reference of weather condition in cities and places of interest for tours.

Usually, visibility is measured by transilluminators or foreword scattering sensors. The basic of the latter device is measuring a small volume of air scattering of light and calculating visibility directly using the relation between scattered light intensity and the extinction coefficient of the atmosphere [1]. However, these two devices do not work well in non-uniform atmosphere. Moreover, their installation and debugging are complicated and expensive in the large scale of application.

There are still several methods to measure visibility from images directly without using devices. The dual-target approach [2] does not take into account the inherent (reference) contrast. In this method, two targets are set up: one is a certain distance from the other. Another approach mentioned in Literature [2, 3] is based edge

information and makes use of the relationship among adjacent pixels to evaluate local contrast. The relations between contrast and target distance, and visibility and contrast are found by curve fitting. Relative visibility [4] can be used to measure daytime visibility as well. The relative visibility is obtained by contrasting and analyzing images at the same position, but in different weather. All these methods of visibility measurement are based analysis of image contrast. Users need to select scene carefully and set targets on purpose. In addition, they are instable and have lower accuracy.

Recently, Hautiere et al. proposed an approach for road visibility measurement using camera calibration and visual analysis [5, 6, 7]. It mainly includes measuring techniques based camera calibration and road vision, and based contrast analysis and stereo vision depth [8]. Although the approach is a convenient measuring technology, it has some limitations: (1) a camera needs to be calibrated precisely in advance. With the change of position and angle, the camera requires re-calibration and adjustment; (2) it is only suitable to measure visibility of regularly shaped roads, otherwise visual model of road can not be created properly, and estimation of the horizontal line's position are more difficult; (3) it is just fit for visibility measurement in foggy weather.

In this paper, we introduced a method of visibility calculation under development. It is based on machine learning and uses video cameras that have already been installed. Contrast is selected as the image feature according to edge information, texture changes and visual features of the captured images. Contrast is a measurement of the different brightness levels between the brightest white and the darkest black in image areas. In other word, it is the difference in color and light between parts of an image and represents the clarity of images. Contrast can be calculated conveniently using active window and multiple spatial scales. There is a certain relationship between contrast and visibility.

The rest of the paper is organized as follows. Section 2 simply describes basic principles of measuring visibility from images. And our proposed image based visibility learning and measuring is explained in Section 3. Section 4 describes some experiments of visibility calculation. Finally, some conclusions and discussions are drawn in Section 5.

2   Principle of Measuring Visibility

Visibility mentioned in this paper is horizontal visibility-how far you can see looking straight ahead. It is meteorologically defined as the maximum distance through the atmosphere toward the horizon at which prominent objects can be identified with the naked eye. We replace visibility by visual range in actual measurement.

For an obtained image, denote the luminance values of target and its horizontal background by B_t and B_h respectively, and then the contrast of target and background, C, is defined as

$$C = (B_t - B_h) / B_h \quad (1)$$

Besides, the contrast satisfies the equation [9]

$$C / C_0 = e^{-\sigma d} \quad (2)$$

where, C_0, a relative constant, is the inherent luminance contrast of target and background; C, the contrast observed from distance d; σ, the extinction coefficient of the atmosphere. Visual range V is defined as the distance between object and observer when the ratio of the two contrasts reduced to ε which is a physical quantity related to human vision, then

$$\varepsilon = e^{-\sigma V} \tag{3}$$

From formula (3), visual range *V* satisfies the following equation directly:

$$V = \frac{-\ln \varepsilon}{\sigma} \tag{4}$$

It is based on Koschmieder law [10]. Above all, there are two ways of visibility. One is to measure σ using some instruments and calculate visibility according to (4) directly; the other, based on visibility definition, is to contrast and analyze the luminance contrast of target and background using image processing technology to measure visibility. Given that the fist approach needs instrumental assistance, the second are selected in our research and its basic principle is introduced in following section.

International meteorological organization recommends ε=0.02; while the International Commission on Illumination suggests ε = 0.05 from a practical standpoint [11]. In fact, the value is always determined through many comparative experiments in practical observation. We suppose ε =0.05 and deduce from formula (4) that,

$$V = \frac{3.0}{\sigma} \tag{5}$$

From formula (2), we obtained

$$\sigma = -\frac{1}{d}\ln \frac{C}{C_0} = \frac{1}{d}\ln \frac{C_0}{C} \tag{6}$$

According formula (5) and (6), visibility can be calculated with the following equation.

$$V = d \cdot 3.0 \Big/ \ln \frac{C_0}{C} \tag{7}$$

For a single observation, the target is fixed, namely distance *d* is a constant. Thus, the link between contrast and visibility is obviously. Traditionally, contrast is calculated according to luminance change of the whole image. However, local contrast, which uses edge information to compute contrast for each pixel, is adopted in our research. It is more suitable for human vision and more close to original definition of visibility as human visual system mainly uses edge information to analyze and recognize objects [2, 3].

3   Measuring Visibility Based on Image Analysis and Learning

The first step is the image preprocessing. Images captured in actual observation always have some problems. The most common situation of image is that the target is fuzzy and many obstacles are contained. In order to quickly locate the target, we have to select a region of interest (ROI). The target in the ROI should have obvious edge

information; the target and the background, large difference in brightness. Only the ROI is analyzed and processed in our research. Images usually contain noise or irrelevant information as well. Therefore, image preprocessing has to be performed first to improve image quality. Noise is mainly produced by the sensor and circuitry of a scanneror digital camera. It also can originate in film grain and in the unavoidable shot noise of an ideal photon detector. The origin of noise determines the distribution characteristic of noise, which is used to choose preprocessing methods.

3.1   Measurement of Local Contrast

According to formula (7), V is a function of d, C_0 and C.

$$V = f(d, C_0, C) \tag{8}$$

As mentioned before, d is a constant. If we suppose that there is an inherent relationship between C_0 and C,

$$C_0 = g(C) \tag{9}$$

V becomes a function of contrast C.

$$V = h(C) \tag{10}$$

There are several different definitions of contrast. One of the most famous definitions is Michelson contrast. It is often used in psychophysics [6] and introduced to quantify the visibility of sinusoidal gratings.

$$C_M = \frac{L\max - L\min}{L\max + L\min} \tag{11}$$

where $L_{\max}$ and $L_{\min}$ are respectively the maximal and the minimal luminance values of the image.

In Weber's theory, if L is the environmental luminance and $\triangle L$ is the slightest variation of the environmental luminance that can be sensed,

$$C_W = \Delta L / L \tag{12}$$

The definitions are both based on luminance. Michelson' is suitable for sinusoidal gratings; Weber's, for targets evenly distributed. However, when stimuli are more complex, neither of them perform well. Moreover, they are not suitable to estimate the contrast in spontaneous images either [6].
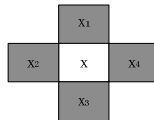
Fig. 1. Adjacent pixels of x

In order to overcome these shortages, we adopt a normalized contrast definition of adjacent pixels based on Weber's.

$$C(x, x_i) = \frac{|f(x) - f(x_i)|}{\max(f(x_i), f(x))} \qquad (13)$$

where, x and x_i are pixels; $f(x)$ and $f(x_i)$ are corresponding pixel gray values. As shown in Fig. 1, there are four adjacent pixels for a pixel x, denoted x_i ($i = 1, 2, 3, 4$). We compute contrast $C(x, x_i)$ using formula (13) for each adjacent pixel x_i, and select the maximum of $C(x, x_i)$ as the contrast of x. Thus, for an ROI, the local contrast of every pixel can be obtained and the number of contrast is the same as that of pixels. If all these contrast is used as image feature, curse of dimensionality will happen. In order to decrease dimensionality, we apply the 4×4 sub-blocks. It means that an ROI is divided into 16 blocks, and, to each block, we calculate the mean of local contrasts of pixels contained in the block. Then, the image feature is reduced to 16-dimensional.

3.2 Supervised Learning for Visibility Measuring

There are many methods to determine the mathematical function of contrasts and visibility. In this paper, a Supported Vector Regression (SVR) based learning system is constructed between image features and the target visibility. It is a kind of supervised learning. The most important for supervised learning is sufficient, typical and accurate training data. A large number of images with exact visibility value must be supplied in training process. They are captured in various scenarios and conditions. The visibility values are established by manual observation or instrumental measurement, for example, using transilluminators or foreword scattering sensors.
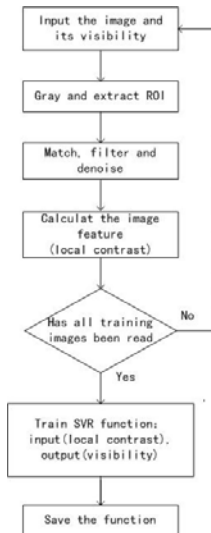
Fig. 2. The flow of training process.   **Fig. 3.** The flow of testing process.
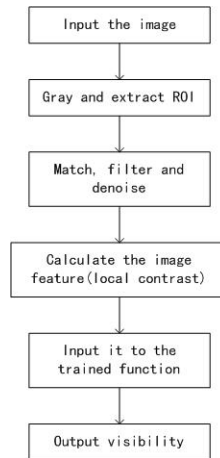
According to practical application, the effectiveness of the relationship between image features and visibility directly influences the accuracy and robustness of the

visibility measuring system. In addition, the learning of data is very large-scaled. Considering the above states, SVR is just suitable for our research.

Support vector machine (SVM) is a computer algorithm that learns by example to assign labels to objects. In essence, SVM maps date to a sufficiently high dimension by an appropriate nonlinear and constructs an optimal separating hyperplane in the high dimension. Many kernel mapping functions can be used in SVM, but a few kernel functions have been found to work well in for a wide variety of applications, such as Radial Basis Function (RBF), Polynomial and Sigmoid. With different kernel functions [12, 13], SVM can be used in various areas, such as dichotomies, multiple-classification and function fitting. SVR is the SVM for regression.

With local contrasts as the image features and visibility values as output, SVR builds a mathematical function of contrast and visibility. Thus, the visibilities of images can be calculated directly. The training flow of SVR is shown in Fig. 2. First, input an image and calculate local contrasts to extract image feature. Second, input the feature to the mathematical function. At last, output the visibility in real time. Fig. 3 demonstrates the test flow.

4   Experiments

The images in our database were all captured by a meteorological observation station which is located in the southern suburb of Beijing. They were captured during Nov. 2010, Dec. 2010 and Jan. 2011. The position in images demonstrated in Fig. 4 is Yizhuang Bridge. As shown Fig. 4, we grayed the image and selected unalterable part of the image as the ROI and then, chosen the parts with obvious edge information as the target of visibility measurement. In other words, the target consisted of chimneys, street lamps and building roofs.
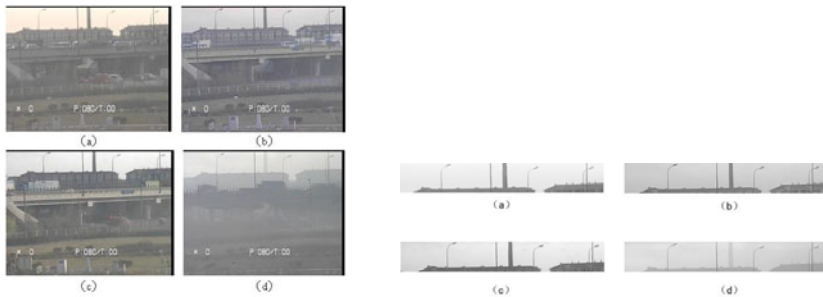
Fig. 4. Caputured images of Yizhuang Bridge and their ROIs: (a) under morning,(b)under nightfall,(c)under sunny weather,(d)under foggy weather

Experiments were performed in chronological order from 23rd December 2010 to 6th January 2011. Training samples of a test were the images captured before the test date and SVR was used to calculate visibility values. The specific steps of a test includes: first, gray images, select the ROI and preprocess (mainly denoise); second, choose the ROI of an image   with clear outline as the standard and transform the ROIs of other images to the standard; at last, divide ROIs into 4×4 blocks, calculate the mean local contrast for each block and construct image features, 16-dimensional contrasts.

Considering that the images captured before 9:00 or after 15:00 usually had low visibility, vague targets and fuzzy backgrounds, we chosen the images obtained between 9:00 and 15:00 as test samples. In addition, we got rid of the images with great errors in training set. SVR was used to train and test. It applied v-SVR and RBF as its kernel function.

Using the above method, test result is showed in TABLE 1 and TABLE 2. Column: Time shows the execution time of the test, training samples, the number of training samples for the current test; test samples, the number of test samples used in the test; success, the number of samples whose visibilities had been calculated within 20% relative errors; Ratio, the percentage of successfully measured test samples.

Table 1. Testing results

Time	Training samples	Test samples	Success	Ratio (%)
2010.12.23	1738	30	30	100
2010.12.24	1783	51	47	92.16
2010.12.25	1858	53	46	86.79
2010.12.26	1911	32	21	65.63
2010.12.27	1961	53	43	81.13
2010.12.28	2042	50	47	94
2010.12.29	2103	41	15	36.59
2010.12.30	2162	53	41	77.36
2010.12.31	2232	39	33	84.62
2011.1.1	2286	53	51	96.23
2011.1.2	2363	53	16	30.19
2011.1.3	2430	53	51	96.23
2011.1.4	2501	53	51	96.23
2011.1.5	2575	53	45	84.91
2011.1.6	2657	53	52	98.11

Table 2. Test Result in January

Time	Training samples	Test samples	Success	Ratio (%)
2011.1.1	2286	53	51	96.23
2011.1.3	2286	53	51	96.23
2011.1.4	2286	53	52	98.11
2011.1.5	2286	53	45	84.91
2011.1.6	2286	53	52	98.11

In the fifteen days stated in TABLE 1, there are seven days on which the percentages of success are above 90% and eleven days above 80%; only two days have the percentages below 40%. The poor percentage should due to low visibilities and fuzzy targets in part time of the tested day. Low visibilities and fuzzy targets usually result in large deviation in calculation.

Moreover, the reference values of visibilities were measured by forward scatter sensor. The sensor has small sampling area, but camera used in our research has large wide-angle and big region. Thus, there are much more difference in visibility between reference values and measured values.

For TABLE 2, the images captured in Dec. 2010 and Nov. 2010 were selected as training samples, not containing Jan. 2011; 1st January 2011 to 6th January 2011, as testing samples. Considering bad image quantity, the test on 2nd January was cancelled.

5   Conclusions

In our research, the images containing target were captured by video cameras, and supervised image learning and visibility real-time calculation were both realized on computer. When the target has clear outline and weather changes less severely, the visibility measuring system basically satisfies the demands of users. It is proved that the methods applied in our research are effective to measure visibility. The system uses video cameras that have already been installed and calculate visibility in real time. It also makes up the defects of existing instruments when local weather has changed in visibility measurement, such as small sampling area, high errors and so on. It is convenient, functional and with lower cost. In one word, it is worth promoting.

References

1. Rossum, M.V., Nieuwenhuizen, T.: Multiple scattering of classical waves: microscopy, mesoscopy and diffusion. Rev. Mod. Phys. 71, 313–371 (1999)
2. Kwon, T.M.: An automatic visibility measurement system based on video cameras. Technical Report, Minnesota Department of Transportation. MN/RC- 1998-25 (1998)
3. Kwon, T.M.: Video camera-based visibility measurement system. United States Patent, Patent No.: US6853453 (2005)
4. Kwon, T.M.: Automatic visibility measurements using video cameras: relative visibility. Technical Report, Minnesota Department of Transportation, CTS-04-03 (2004)
5. Hautiere, N., Tarel, J.-P., Lavenant, J., Aubert, D.: Automatic fog detection and estimation of visibility distance through use of an onboard camera. Machine Vision and Applications 17(1), 8–20 (2006)
6. Hautiere, N., Labayrade, R., Aubert, D.: Real-time disparity contrast combination for onboard estimation of the visibility distance. IEEE Trans. Intelligent Transportation Systems 7(2), 201–212 (2006)
7. Hautiere, N., Aubert, D., Dumont, E., Tarel, J.-P.: Experimental validation of dedicated methods to in-vehicle estimation of atmospheric visibility distance. IEEE Trans. Instrumentation and Measurement 57(10), 2218–2225 (2008)
8. Saxena, A., Chung, S.H., Ng, A.Y.: 3D depth reconstruction from a single still image. International Journal of Computer Vision 76(1), 53–69 (2008)
9. Duntley, S.Q.: The reduction of apparent contrast by the atmosphere. J. Opt. Soc. Am. 38, 179–191 (1948)
10. Middleton, W.E.K.: Vision through the atmosphere, vol. 64. University of Toronto Press, Toronto (1952)
11. Dumont, E., Cavallo, V.: Extended photometric model of fog effectson road vision. Transp. Res. Rec.: J. Transp. Res. Board (1862), 77–81 (2004)
12. Schlkopf, B., Burges, C., Vapnik, V.: Extracting support data for a given task. In: Fayyad, U.M., Uthurusamy, R. (eds.) Proc. of First Intl. Conf. on Knowledge Discovery & Data Mining, pp. 262–267. AAAI Press (1995)
13. Vapnik, V., Golowich, S., Smola, A.: Support vector method for function approximation, regression estimation, and signal processing. In: Mozer, M., Jordan, M., Petsche, T. (eds.) Neural Information Processing Systems. MIT Press (1997)

Polynomial Time Algorithm for Learning Globally Optimal Dynamic Bayesian Network

Nguyen Xuan Vinh[1], Madhu Chetty[1], Ross Coppel[2], and Pramod P. Wangikar[3]

[1] Gippsland School of Information Technology, Monash University, Australia
[2] Department of Microbiology, Monash University, Australia
[3] Chemical Engineering Department, Indian Institute of Technology, Mumbai, India
{vinh.nguyen,madhu.chetty,Ross.Coppel}@monash.edu, wangikar@iitb.ac.in

Abstract. This paper is concerned with the problem of learning the globally optimal structure of a dynamic Bayesian network (DBN). We propose using a recently introduced information theoretic criterion named MIT (Mutual Information Test) for evaluating the goodness-of-fit of the DBN structure. MIT has been previously shown to be effective for learning static Bayesian network, yielding results competitive to other popular scoring metrics, such as BIC/MDL, K2 and BD, and the well-known constraint-based PC algorithm. This paper adapts MIT to the case of DBN. Using a modified variant of MIT, we show that learning the globally optimal DBN structure can be efficiently achieved in polynomial time.

Keywords: Dynamic Bayesian network, global optimization, gene regulatory network.

1   Introduction

Bayesian network (BN) is a central topic in machine learning, and has found numerous applications [8]. Two important disadvantages when applying the traditional static BN model to certain domain problems, such as gene regulatory network reconstruction in bioinformatics, are: *(i)* BN does not have a mechanism for exploiting the temporal aspect of time-series data; and *(ii)* BN does not allow the modeling of cyclic phenomena, such as feed back loops, which are prevalent in biological systems [13, 9]. These drawbacks have motivated the development of the so-called dynamic Bayesian network (DBN). The simplest model of this type is the first-order Markov stationary DBN, in which both the structure of the network and the parameters characterizing it are assumed to remain unchanged over time, as exemplified in Fig. 1a. In this model, the value of a random variable (RV) at time $t+1$ is assumed to depend only on the value of its parents at time t. DBN accounts for the temporal aspect of time-series data, in that an edge must always direct forward in time, and allows feedback loops (Fig. 1b). Since its inception, DBN has received particular interest, especially from the bioinformatics community [7, 13, 14, 12]. Recent works in the machine learning community have progressed to allow more flexible DBN models, such as

one with, either parameters [5], or both structure and parameters [10,4] changing over time. It is worth noting that more flexible models generally require more data to be learned accurately. In situations where training data are scarce, such as in microarray experiments where the data size can be as small as a couple of dozen samples, a simpler model such as the first-order Markov stationary DBN might be a more suitable choice.
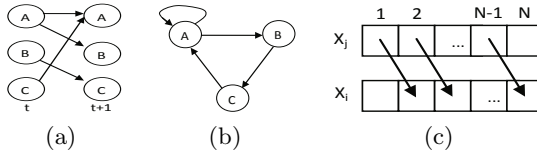
Fig. 1. Dynamic Bayesian Network: (a) 1st order Markov stationary DBN; (b) its equivalent folded network; (c) data alignment for dynamic Bayesian network with an edge $X_j \rightarrow X_i$. The "effective" number of observations is now only $N - 1$.

In this paper, we focus on the problem of learning the globally optimal structure for the first-order Markov stationary DBN. Henceforth, DBN shall refer to this particular class of stationary DBN, and *learning* shall refer to *structure learning*. The most popular approaches for learning DBN have been the ones adapted from the static BN literature, namely the *search+score* paradigm [13,12], and Markov Chain Monte Carlo (MCMC) simulation [7,4,10]. In this paper, we are interested in the *search+score* approach, in which we specify a scoring function to assess the goodness-of-fit of a DBN given the data, and a search procedure to find the optimal network based on this scoring metric. Several popular scores for static BN, such as the Bayesian scores (K2, Bayesian-Dirichlet (BD), BDe and BDeu), and the information theoretic scores (Bayesian Information Criterion (BIC)/minimal description length (MDL) and Akaike Information Criterion—AIC), can be adapted straightforwardly for DBN. Another recently introduced scoring metric that catches our interest is the so-called MIT (Mutual Information Test) score [1], which, as the name suggests, belongs to the family of scores based on information theory. Through extensive experimental validation, the author suggests that MIT can compete favorably with Bayesian scores, outperforms BIC/MDL and should be the score of reference within those based on information theory. As opposed to the other popular scoring metrics, MIT has not been considered for DBN learning to our knowledge.

As for the *search* part, due to several non-encouraging complexity results (i.e., NP-hardness) in learning static BN [2], most authors have resorted to heuristic search algorithms when it comes to learning DBN. Recently, Dojer [3] has shown otherwise that learning DBN structure, as opposed to static BN, does not necessarily have to be NP-hard. In particular, this author showed that, under some mild assumptions, there are algorithms for finding the globally optimal network with a polynomial worst-case time complexity, when the MDL and BDe scores are used. In the same line of these findings, in this paper, we shall show

that there exists a polynomial worst-case time complexity algorithm for learning the globally optimal DBN under the newly introduced MIT scoring metric. Our experimental results show that, in terms of the recovered DBN quality, MIT performs competitively with BIC/MDL and BDe. In terms of theoretical complexity analysis, globalMIT admits a comparable worst-case complexity to the BIC/MDL-based global algorithm, and is much faster than the BDe-based algorithm. The paper is organized as follows: in Section 2 we review the MIT score for DBN learning. Section 3 presents our algorithm for finding the globally optimal network, followed by experimental results in section 4.

2   MIT Score for Dynamic Bayesian Network Learning

Let us first review the MIT score for learning BN, which can then be adapted to the DBN case. Briefly speaking, under MIT the goodness-of-fit of a network is measured by the total mutual information shared between each node and its parents, penalized by a term which quantifies the degree of statistical significance of this shared information. Let $\mathbf{X} = \{X_1, \ldots, X_n\}$ denote the set of n variables with corresponding $\{r_1, \ldots, r_n\}$ discrete states, D denote our data set of N observations, G be a DAG, and $\mathbf{Pa}_i = \{X_{i1}, \ldots, X_{is_i}\}$ be the set of parents of X_i in G with corresponding $\{r_{i1}, \ldots, r_{is_i}\}$ discrete states, $s_i = |\mathbf{Pa}_i|$, then the MIT score is defined as:

$$SS_{MIT}(G : D) = \sum_{i=1; \mathbf{Pa}_i \neq \emptyset}^{n} \{2N \cdot I(X_i, \mathbf{Pa}_i) - \sum_{j=1}^{s_i} \chi_{\alpha, l_{i\sigma_i(j)}}\}$$

where $I(X_i, \mathbf{Pa}_i)$ is the mutual information between X_i and its parents as estimated from D. $\chi_{\alpha, l_{ij}}$ is the value such that $p(\chi^2(l_{ij}) \leq \chi_{\alpha, l_{ij}}) = \alpha$ (the Chi-square distribution at significance level $1 - \alpha$), and the term $l_{i\sigma_i(j)}$ is defined as:

$$l_{i\sigma_i(j)} = \begin{cases} (r_i - 1)(r_{i\sigma_i(j)} - 1) \prod_{k=1}^{j-1} r_{i\sigma_i(k)}, & j = 2 \ldots, s_i \\ (r_i - 1)(r_{i\sigma_i(j)} - 1), & j = 1 \end{cases}$$

where $\sigma_i = \{\sigma_i(1), \ldots, \sigma_i(s_i)\}$ is any permutation of the index set $\{1 \ldots s_i\}$ of $\mathbf{Pa}_i$, with the first variable having the greatest number of states, the second variable having the second largest number of states, and so on. It can be shown that the mutual information part of the score is equivalent to the log-likelihood score, while the second part serves as a penalty term. For detailed motivations and derivation of this scoring metric as well as an extensive comparison with BIC/MDL and BD, we refer readers to [1].

Adapting MIT for DBN learning is rather straightforward. Essentially, the mutual information is now calculated between a parent set and its child, which should be 1-unit shifted in time, as required by the first-order Markov assumption, denoted by $X_i^{\overrightarrow{1}} = \{X_{i2}, X_{i3}, \ldots, X_{iN}\}$. As such, the number of "effective" observations, denoted by N_e, for DBN is now only $N - 1$. Similarly, when the data is composed of N_t separate time-series, the number of effective observations is only $N_e = N - N_t$. This is demonstrated in Figure 1(c). The MIT score for DBN should be calculated as:

$$S'_{MIT}(G : D) = \sum_{i=1; \mathbf{Pa}_i \neq \emptyset}^{n} \{2N_e.I(X_i^{\overrightarrow{1}}, \mathbf{Pa}_i) - \sum_{j=1}^{s_i} \chi_{\alpha, l_{i\sigma_i(j)}}\}$$

3   Optimal Dynamic Bayesian Network Structure Learning in Polynomial Time with MIT

In this section, we show that learning the globally optimal DBN with MIT can be achieved in polynomial time. Our development is based on a recent result presented in [3], which states that under several mild assumptions, there exists a polynomial worst-case time complexity algorithm for learning the optimal DBN with the MDL and BDe scoring metrics. Specifically, the 4 assumptions that Dojer [3] considered are:

Assumption 1. *(acyclicity) There is no need to examine the acyclicity of the graph.*

Assumption 2. *(additivity)* $S(G : D) = \sum_{i=1}^{n} s(X_i, \boldsymbol{Pa}_i : D|_{X_i \cup \boldsymbol{Pa}_i})$ *where* $D|_{X_i \cup \boldsymbol{Pa}_i}$ *denotes the restriction of D to the values of the members of* $X_i \cup \boldsymbol{Pa}_i$.

To simplify notation, we write $s(\mathbf{Pa}_i)$ for $s(X_i, \mathbf{Pa}_i : D|_{X_i \cup \mathbf{Pa}_i})$.

Assumption 3. *(splitting)* $s(\boldsymbol{Pa}_i) = g(\boldsymbol{Pa}_i) + d(\boldsymbol{Pa}_i)$ *for some non-negative functions g,d satisfying* $\boldsymbol{Pa}_i \subseteq \boldsymbol{Pa}'_i \Rightarrow g(\boldsymbol{Pa}_i) \leq g(\boldsymbol{Pa}'_i)$

Assumption 4. *(uniformity)* $|\boldsymbol{Pa}_i| = |\boldsymbol{Pa}'_i| \Rightarrow g(\boldsymbol{Pa}_i) = g(\boldsymbol{Pa}'_i)$

Assumption 1 is valid for DBN in general (since the edges only directs forward in time, acyclicity is automatically satisfied). Assumption 2 states that the scoring function decomposes over the variables, which is obvious for MIT. Together with assumption 1, this assumption allows us to compute the parents set of each variable independently. Assumption 3 requires the scoring function to decompose into two components: d evaluating the accuracy of representing the distribution underlying the data by the network, and g measuring its complexity. Furthermore, g is required to be a monotonically non-decreasing function in the cardinality of $\mathbf{Pa}_i$ (assumption 4).

We note that unlike MIT in its original form that we have considered above, where better networks have higher scores, for the score considered by Dojer, lower scored networks are better. And thus the corresponding optimization must be cast as a score minimization problem. We now consider a variant of MIT as follows:

$$S_{MIT}(G : D) = \sum_{i=1}^{n} 2N_e.I(\overrightarrow{X_i^1}, \mathbf{X}) - S'_{MIT}(G : D) \qquad (1)$$

which admits the following decomposition over each variable (with the convention of $I(X_i, \emptyset) = 0$):

$$s_{MIT}(\mathbf{Pa}_i) = d_{MIT}(\mathbf{Pa}_i) + g_{MIT}(\mathbf{Pa}_i)$$
$$d_{MIT}(\mathbf{Pa}_i) = 2N_e.I(\overrightarrow{X_i^1}, \mathbf{X}) - 2N_e.I(\overrightarrow{X_i^1}, \mathbf{Pa}_i)$$
$$g_{MIT}(\mathbf{Pa}_i) = \sum_{j=1}^{s_i} \chi_{\alpha, l_i \sigma_i(j)}$$

Roughly speaking, d_{MIT} measures the "error" of representing the joint distribution underlying D by G, while g_{MIT} measures the complexity of this representation. It is obvious that the problem of S'_{MIT} maximization is equivalent to the problem of S_{MIT} minimization, since $\sum_{i=1}^{n} 2N_e.I(\overrightarrow{X_i^1}, \mathbf{X}) = const$. Also, it is straight-forward to show that d_{MIT} and g_{MIT} satisfy assumption 3. Unfortunately, g_{MIT} does not satisfy assumption 4. However, for many applications, if all the variables have the same number of states then it can be shown that g_{MIT} satisfies assumption 4.

Assumption 5. *(variable uniformity) All variables in $\mathbf{X}$ have the same number of discrete states k.*

Proposition 1. *Under the assumption of variable uniformity, g_{MIT} satisfies assumption 4.*

Proof. It can be seen that if $|\mathbf{Pa}_i| = |\mathbf{Pa}'_i| = s_i$, then $g_{MIT}(\mathbf{Pa}_i) = g_{MIT}(\mathbf{Pa}'_i) = \sum_{j=1}^{s_i} \chi_{\alpha,(k-1)^2 k^{j-1}}$. □

Since $g_{MIT}(\mathbf{Pa}_i)$ is the same for all parent sets of the same cardinality, we can write $g_{MIT}(|\mathbf{Pa}_i|)$ in place of $g_{MIT}(\mathbf{Pa}_i)$. With assumptions 1-5 satisfied, we can employ the following Algorithm 1, named globalMIT, to find the globally optimal DBN with MIT, i.e., the one with the minimal S_{MIT} score.

Algorithm 1. globalMIT : Optimal DBN with MIT

$\quad \mathbf{Pa}_i := \emptyset$
$\quad \textbf{for } p = 1 \textbf{ to } n \textbf{ do}$
$\quad\quad$ If $g_{MIT}(p) \geq s_{MIT}(\mathbf{Pa}_i)$ then return $\mathbf{Pa}_i$; Stop.
$\quad\quad \mathbf{P} = \arg\min_{\{\mathbf{Y} \subseteq \mathbf{X}: |\mathbf{Y}|=p\}} s_{MIT}(\mathbf{Y})$
$\quad\quad$ If $s_{MIT}(\mathbf{P}) < s_{MIT}(\mathbf{Pa}_i)$ then $\mathbf{Pa}_i := \mathbf{P}$.
$\quad \textbf{end for}$

Theorem 1. *Under assumptions 1-5, globalMIT applied to each variable in $\mathbf{X}$ finds a globally optimal DBN under the MIT scoring metric.*

Proof. The key insight here is that once a parent set grows to a certain extent, its complexity alone surpasses the total score of a previously found sub-optimal parent set. In fact, all the remaining potential parent sets $\mathbf{P}$ omitted by the algorithm have a total score higher than the current best score, i.e., $s_{MIT}(\mathbf{P}) \geq g_{MIT}(|\mathbf{P}|) \geq s_{MIT}(\mathbf{Pa}_i)$, where $\mathbf{Pa}_i$ is the last sub-optimal parent set found. □

We note that the terms $2N_e.I(\overrightarrow{X_i^1}, \mathbf{X})$ in the S_{MIT} score in (1) do not play any essential role, since they are all constant and would not affect the outcome of our optimization problem. Knowing their exact value is however, necessary for the stopping criterion in Algorithm 1, and also for constructing its complexity bound, as we shall do shortly. Unfortunately, calculating $I(\overrightarrow{X_i^1}, \mathbf{X})$ is by itself a hard problem, requiring $O(k^{n+1})$ space and time in general. However, for our purpose, since the only requirement for d_{MIT} is that it must be non-negative,

it is sufficient to use an upper bound of $I(X_i^{\overrightarrow{1}}, \mathbf{X})$. A fundamental property of the mutual information states that $I(\mathbf{X}, \mathbf{Y}) \leq \min\{H(\mathbf{X}), H(\mathbf{Y})\}$, i.e., mutual information is bounded by the corresponding entropies. We therefore have:

$$2N_e.I(X_i^{\overrightarrow{1}}, \mathbf{X}) \leq 2N_e.H(X_i^{\overrightarrow{1}}),$$

where $H(X_i^{\overrightarrow{1}})$ can be estimated straightforwardly from the data. Or else, we can use an a priory fixed upper bound for all $H(X_i^{\overrightarrow{1}})$, that is $\log k$, then:

$$2N_e.I(X_i^{\overrightarrow{1}}, \mathbf{X}) \leq 2N_e.\log k.$$

Using these bounds, we obtain the following more practical versions of d_{MIT}:

$$d'_{MIT}(\mathbf{Pa}_i) = 2N_e.H(X_i^{\overrightarrow{1}}) - 2N_e.I(X_i^{\overrightarrow{1}}, \mathbf{Pa}_i)$$
$$d''_{MIT}(\mathbf{Pa}_i) = 2N_e.\log k - 2N_e.I(X_i^{\overrightarrow{1}}, \mathbf{Pa}_i)$$

It is straightforward to show that Algorithm 1 and Theorem 1 are still valid when d'_{MIT} or d''_{MIT} are used in place of d_{MIT}.

3.1   Complexity Bound

Theorem 2. *globalMIT admits a polynomial worst-case time complexity in the number of variables.*

Proof. Our aim is to find a number p^* satisfying $g_{MIT}(p^*) \geq s_{MIT}(\emptyset)$. Clearly, there is no need to examine any parent set of cardinality p^* and over. In the worse case, our algorithm will have to examine all the possible parent sets of cardinality from 1 to $p^* - 1$. We have:

$$g_{MIT}(p^*) \geq s_{MIT}(\emptyset) \Leftrightarrow \sum_{j=1}^{p^*} \chi_{\alpha, l_i \sigma_i(j)} \geq d_{MIT}(\emptyset) = 2N_e.I(X_i^{\overrightarrow{1}}, \mathbf{X}).$$

As discussed above, since calculating d_{MIT} is not convenient, we use d'_{MIT} and d''_{MIT} instead. With d'_{MIT} and d''_{MIT}, p^* can be found respectively as:

$$p^* = \arg\min\{p | \textstyle\sum_{j=1}^{p} \chi_{\alpha, l_i \sigma_i(j)} \geq 2N_e.H(X_i^{\overrightarrow{1}})\} \tag{2}$$
$$p^* = \arg\min\{p | \textstyle\sum_{j=1}^{p} \chi_{\alpha, l_i \sigma_i(j)} \geq 2N_e.\log k\}. \tag{3}$$

It can be seen that p^* depends only on α, k and N_e. Since there are $O(n^{p^*})$ subsets with at most p^* parents, and each set of parents can be scored in polynomial time, globalMIT admits an overall polynomial worst-case time complexity in the number of variable n.   □

We now give some examples to demonstrate the practicability of Theorem 2.

Example 1: Consider a gene regulatory network reconstruction problem, where each gene has been discretized to $k = 3$ states, corresponding to up, down and regular gene expression. With the level of significance α set to 0.999 as recommended in [1], we have $g_{MIT}(1) = 18.47$; $g_{MIT}(2) = 51.37$; $g_{MIT}(3) = 119.35 \ldots$ Consider a data set of $N = 12$ observations, which is the popular length of microarray time-series experiments (in fact N often ranges within $4-15$), then $d''_{MIT}(\emptyset) = 2(N-1)\log k = 24.16$. Observing that $g_{MIT}(2) > d''_{MIT}(\emptyset)$, then $p^* = 2$ and we do not have to consider any parent sets of 2 variables or more. Let us compare this bound with those of the algorithms for learning the globally optimal DBN under the BIC/MDL and BDe scoring metrics. For BIC/MDL, p^*_{MDL} is given by $\lceil \log_k N \rceil$, while for BDe, $p^*_{BDe} = \lceil N \log_{\gamma^{-1}} k \rceil$, where the distribution $P(G) \propto \gamma^{\sum |\mathbf{Pa}_i|}$, with a penalty parameter $0 < \gamma < 1$, is used as a prior over the network structures [3]. In this case, $p^*_{MDL} = 3$. If we choose $\log \gamma^{-1} = 1$ then $p^*_{BDe} = \lceil N \log k \rceil = 14$. In general, p^*_{BDe} scales linearly with the number of data items N, making its value less of practical interest, even for small data sets.

Example 2: Since the number of observations in a single microarray time-series experiment is often limited, it is a popular practice to concatenate several time-series to obtain a larger data set for analysis. Let us merge $N_t = 10$ data sets, each with 12 observations, then $N_e = N - N_t = 120 - 10 = 110$. For this combined data set, $g_{MIT}(4) > d''_{MIT}(\emptyset) = 2N_e \log k = 241.69 \Rightarrow p^* = 4$, thus there is no need to consider any parent set of more than 3 variables. Of course, this analysis only gives us the worst-case time complexity. In practice, the execution of Algorithm 1 can often be much shorter, since $s_{MIT}(\mathbf{Pa}_i)$ is often much greater than $s_{MIT}(\emptyset)$. For comparison, we have $p^*_{MDL} = 5$, and $p^*_{BDe} = 132$ with $\log \gamma^{-1} = 1$.

4   Experimental Evaluation

We next describe our experiments to evaluate our global approach for learning DBN with MIT, and compare it with the other most popular scores, namely BIC/MDL and BD. Our method, implemented in Matlab, was used, along with BNFinder [12], a Python-based software for inferring the globally optimal DBN with the MDL and BDe scores as proposed by Dojer [3]. In addition, we also employed the Java-based Banjo software [6], which can perform greedy search and simulated annealing over the DBN space using the BDeu metric. The specific problem domain that we shall work with in this experiment is the problem of gene regulatory network reconstruction from microarray data, with the variables being genes, and edges being regulatory relationship between genes. We employ several synthetic data sets generated by different data generation schemes that have been used in some previous studies, namely, probabilistic method [7], linear dynamical system based method [13], and non-linear dynamical system based method [11]. As a realistic number of samples for microarray data, we generated data sets of between 30 and 300 samples. With the ground-truth network available, we count the number of true positive (TP), false positive (FP), true negative (TN)

and false negative (FN) edges, and report two network quality metrics, namely *sensitivity*= TP/(TP+FN), and *imprecision*=FP/(FP+TP).

Parameters setting: globalMIT has one parameter, namely the significance level α, to control the trade-off between goodness-of-fit and network complexity. Adjusting α will generally affect the sensitivity and imprecision of the discovered network, very much like its affect on the Type-I and Type-II error of the mutual information test of independence. de Campos [1] suggested using very high levels of significance, namely 0.999 and 0.9999. We note that, the data sets used in [1] are of sizes from 1000 to 10000 samples. For microarray data sets of merely $30 - 300$ samples, it is necessary to use a lower level of significance α to avoid overly penalizing network complexity. We have experimentally observed that using $\alpha \in [0.95, 0.999]$ on these small data sets yielded reasonable results, with balanced sensitivity and imprecision. BNFinder+MDL required no parameter tuning, while for BNFinder+BDe, the pseudo-counts for the BDe score was set to the default value of 1, and the penalty parameter was set to the default value of $\log \gamma^{-1} = 1$. For Banjo, we employed simulated annealing as the search engine, and left the equivalent sample size to the default value of 1 for the BDeu score, while the max-fan-in was set to 3. The runtime for Banjo was set to the average runtime of globalMIT, with a minimum value of 10 minutes, in case where globalMIT terminates earlier. Since some experiments were time consuming, all our experiments were performed in parallel on a 16-core Xeon X5550 workstation.

Probabilistic Network Synthetic Data: We employed a subnetwork of the yeast cell cycle, consisting of 12 genes and 11 interactions, as depicted in Fig. 2(a). Two different conditional probabilities were associated with these interactions, namely noisy regulation according to a binomial distribution, and noisy XOR-style co-regulation (see [7] for the parameter details, and this author website for Matlab code to generate this data). In addition, 8 unconnected nodes were added as confounders, for a total of 20 nodes. For each number of samples $N = 30, 70$ and 100, we generated 10 data sets. From the average statistics in Table 1, it can be seen that this is a relatively easy case for all methods. Except Banjo which committed a lower sensitivity and yet a higher imprecision, all other methods nearly recovered the correct network. Note that due to the excessive runtime of BNFinder+BDe, for $N = 100$, only 1 of ten data sets was analyzed.

(a) The yeast cell cycle   (b) Yu's net No. 1   (c) Yu's net No. 2
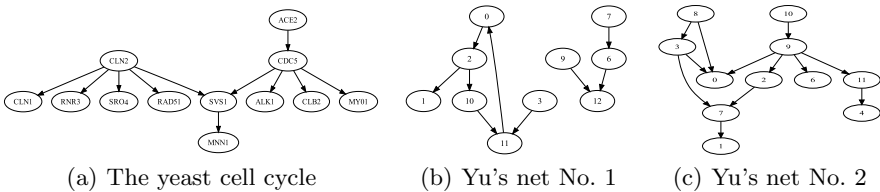
Fig. 2. Synthetic Dynamic Bayesian Networks

Linear Dynamical System Synthetic Data: We employed the two synthetic networks as described in [13], each consisting of 20 genes, with 10 and 11 genes having regulatory interaction, while the remainder moving in a random walk, as depicted in Fig. 2(b,c). The data are generated by a simple linear dynamical process as: $X_{t+1} - X_t = A(X_t - T) + \epsilon$, with X denoting the expression profiles, A describes the strength of gene-gene regulations, T is the constitutive expression values, and ϵ simulates a uniform biological noise. The detailed parameters for each network can be found in [13]. Using the GeneSim software provided by these authors, we generated, for each number of sample $N = 100, 200$ and 300, 10 data sets for each network. From the average statistics in Table 1, it can be seen that Banjo performed well on both data sets. BNFinder achieved a slightly lower sensitivity, but with very high imprecision rate. It is probable that the self-link suppression default option in BNFinder has led the method to include more incorrect edges to the network for a reasonable goodness-of-fit. GlobalMIT performed worse at $N = 100$, but is better at higher number of samples. Again, due to time limit, we were only able to run BNFinder+BDe on one out of ten data sets for each network at $N = 200$ and 300.

Table 1. Experimental Results

N	GlobalMIT Sen	Imp	Time	Banjo Sen	Imp	Time	BNFinder+MDL Sen	Imp	Time	BNFinder+BDe Sen	Imp	Time
Probabilistic Network Synthetic Data												
30	95 ± 9	29 ± 13	13 ± 3	84 ± 6	70 ± 4	600	86 ± 10	10 ± 9	< 2	85 ± 8	11 ± 11	52 ± 4
70	100 ± 0	1 ± 3	67 ± 4	82 ± 0	51 ± 6	600	100 ± 0	5 ± 7	25 ± 1	100 ± 0	3 ± 4	2.7 ± 0.5h
100	100 ± 0	0 ± 0	499 ± 56	82 ± 0	43 ± 2	600	100 ± 0	1 ± 3	34 ± 1	100	0	9.4h*
Linear Dynamical System Synthetic Data: Yu's net No. 1												
100	54 ± 12	54 ± 13	66 ± 5	58 ± 9	35 ± 16	600	58 ± 9	72 ± 4	4 ± 1	67 ± 7	74 ± 4	4.4 ± 1.3h
200	77 ± 4	19 ± 9	409 ± 127	67 ± 5	8 ± 9	600	66 ± 4	74 ± 2	47 ± 5	67	84	13.6h*
300	79 ± 4	19 ± 12	$.6 \pm .07$h	69 ± 7	4 ± 6	0.6h	68 ± 4	77 ± 2	49 ± 5	67	84	26.5h*
Linear Dynamical System Synthetic Data: Yu's net No. 2												
100	22 ± 15	72 ± 17	44 ± 8	38 ± 11	59 ± 13	600	28 ± 12	83 ± 7	3 ± 1	30 ± 16	86 ± 7	3.3 ± 0.8h
200	49 ± 15	35 ± 19	534 ± 158	45 ± 14	37 ± 16	600	38 ± 8	79 ± 4	39 ± 5	42	85	12.1h*
300	62 ± 12	24 ± 11	$.49 \pm .05$h	53 ± 9	17 ± 13	0.49h	47 ± 9	78 ± 4	40 ± 6	50	85	21.2h*
Non-Linear Dynamical System Synthetic Data												
99	37 ± 10	59 ± 12	< 1	7 ± 3	13 ± 32	600	13 ± 11	81 ± 14	< 1	16 ± 13	77 ± 17	< 1
150	39 ± 16	58 ± 16	< 1	9 ± 11	16 ± 35	600	19 ± 15	71 ± 23	< 1	24 ± 18	67 ± 24	< 1
300	61 ± 7	51 ± 6	< 1	10 ± 12	30 ± 48	600	24 ± 14	74 ± 14	< 1	23 ± 20	80 ± 15	< 1

Sen: percent sensitivity; Imp: percent imprecision; Time: in seconds, unless otherwise specified
*: only run on one data set.

Non-Linear Dynamical System Synthetic Data: We employed a five-gene network as in [11], of which dynamics is modeled by a system of coupled differential equations adhering to the power-law formalism, called the S-system. The concrete form of an S-system is given as follows:

$$\frac{dX_i}{dt} = \alpha_i \prod_{j=1}^{n} X_j^{g_{ij}} - \beta_i \prod_{j=1}^{n} X_j^{h_{ij}}, \quad i = 1 \ldots n, \tag{4}$$

where the rates α_i, β_i and kinetic orders g_{ij} and h_{ij} are parameters dictating the influence of gene X_j on the rate of change in the expression level of gene

X_i. Using the same system parameters as in [11], we integrated the system using the Runge-Kutta method with 10 different initial conditions to obtain 10 time series, each of length 50. We then randomly chose 3 time series of length 33, 3 of length 50 and 6 of length 50, to make data sets of length $N = 99, 150$ and 300 respectively, with 10 data sets for each N value. Although this data had been previously analyzed with good accuracy by using differential equation models, it proved to be the most challenging case for DBN based methods. Even with a fairly large number of samples, compared to a small number of variables and interactions, all the methods performed poorly, with low sensitivity and high imprecision, rendering the results hardly useful. GlobalMIT nevertheless showed a slight advantage, with a reasonable sensitivity and imprecision at $N = 300$.

5   Conclusion

This paper has investigated the problem of learning the globally optimal DBN structure with the MIT scoring metric. We have showed that this task can be achieved using a polynomial time algorithm. Compared with the other well-known scoring metrics, namely BIC/MDL and BDe, both in terms of the worst-case complexity bound and practical evaluation, the BIC/MDL-based algorithm for learning the globally optimal DBN is fastest, followed by MIT, whereas the extensive runtime required by the BDe-based algorithm renders it a very expensive option. GlobalMIT, which is based on a sound information theoretic criterion, represents a very competitive alternative, both in terms of the network quality and runtime required.

References

1. de Campos, L.M.: A scoring function for learning bayesian networks based on mutual information and conditional independence tests. J. Mach. Learn. Res. 7, 2149–2187 (2006)
2. Chickering, D.M.: Learning Bayesian Networks is NP-Complete. In: Fisher, D., Lenz, H. (eds.) Learning from Data: Artificial Intelligence and Statistics V, pp. 121–130 (1996)
3. Dojer, N.: Learning Bayesian Networks Does Not Have to Be NP-Hard. In: Proceedings of International Symposium on Mathematical Foundations of Computer Science, pp. 305–314 (2006)
4. Dondelinger, F., Lebre, S., Husmeier, D.: Heterogeneous continuous dynamic bayesian networks with flexible structure and inter-time segment information sharing. In: ICML, pp. 303–310 (2010)
5. Grzegorczyk, M., Husmeier, D.: Non-stationary continuous dynamic Bayesian networks. In: NIPS 2009 (2009)

6. Hartemink, A.: Banjo: A structure learner for static and dynamic bayesian networks, http://www.cs.duke.edu/~amink/software/banjo
7. Husmeier, D.: Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks. Bioinformatics 19(17), 2271–2282 (2003)
8. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. The MIT Press (2009)
9. Ram, R., Chetty, M., Dix, T.: Causal modeling of gene regulatory network. In: IEEE CIBCB 2006 (2006)
10. Robinson, J., Hartemink, A.: Learning Non-Stationary Dynamic Bayesian Networks. The Journal of Machine Learning Research 11, 3647–3680 (2010)
11. Sugimoto, N., Iba, H.: Inference of gene regulatory networks by means of dynamic differential bayesian networks and nonparametric regression. Genome Informatics 15(2), 121–130 (2004)
12. Wilczynski, B., Dojer, N.: BNFinder: exact and efficient method for learning Bayesian networks. Bioinformatics 25(2), 286–287 (2009)
13. Yu, J., Smith, V.A., Wang, P.P., Hartemink, A.J., Jarvis, E.D.: Advances to Bayesian network inference for generating causal networks from observational biological data. Bioinformatics 20(18), 3594–3603 (2004)
14. Zou, M., Conzen, S.D.: A new dynamic Bayesian network (DBN) approach for identifying gene regulatory networks from time course microarray data. Bioinformatics 21(1), 71–79 (2005)

A Hybrid FMM-CART Model for Fault Detection and Diagnosis of Induction Motors

Manjeevan Seera[1], CheePeng Lim[2,*], and Dahaman Ishak[1]

[1] School of Electrical and Electronic Engineering
[2] School of Computer Sciences
University of Science Malaysia
`cplim@cs.usm.my`

Abstract. A new approach to detect and classify fault conditions of induction motors using a hybrid Fuzzy Min-Max (FMM) neural network and the Classification and Regression Tree (CART) is proposed. The hybrid model, known as FMM-CART, exploits the advantages of both FMM and CART for undertaking data classification and rule extraction problems. A series of experiments using real data measurements of motor currents from healthy and faulty induction motors is conducted. FMM-CART is able to detect and classify the associated inductor motor faults with good accuracy rates. Useful rules in the form of a decision tree are also elicited from FMM-CART to analyze and understand different fault conditions of induction motors.

Keywords: Fault detection and diagnosis, fuzzy min-max neural network, classification and regression tree, induction motor.

1 Introduction

Electric motors are widely used for converting electrical energy to mechanical energy. In real applications, it is important for electrical motors to have a continuous and safe operation. As such, effective fault detection and diagnosis techniques are needed in order to reduce maintenance and downtime of motors. Among different types of electric motors, induction motors contribute more than 60% of the electrical energy produced [1]. It is reported in [2-3] that shipment of AC motors and three-phase induction motors in Europe are at 96.2% and 87%, respectively. As such, the present study aims at designing an effective and low-cost approach to detecting and classifying fault conditions of induction motors.

The Motor Current Signature Analysis (MCSA) approach, which is a low-cost and convenient method for fault detection and diagnosis, is deployed in this study. MCSA deploys the results of spectral analysis of the supply current to detect a particular motor failure in the drive system. An effective use of MCSA in fault detection of induction motors is to sample the harmonics components in the stator current spectrum using the fast Fourier transform (FFT) [4], which is employed in this study.

In terms of fault diagnosis, rule extraction plays an important role to provide explanation of the predictions to users. It is reported in [5] that early detection of induction motor faults increases the safety of operators handling the machines, and the rules extracted from decision trees and adaptive neuro-fuzzy inference serve to achieve this purpose. In [6], rules for unbalanced supply, unbalanced mechanical load, encoder, and voltmeter failures are extracted from a neuro-fuzzy system.

The organization of this paper is as follows. In Section 2, the proposed FMM-CART model is explained. The experimental study, the results and discussion are included in Section 3.   Finally, concluding remarks are given in Section 4.

2   The Hybrid FMM-CART Model

Two FMM neural network models were proposed by Simpson: one for pattern classification (a supervised learning model) [7] and another for pattern clustering (an unsupervised learning model) [8]. The supervised FMM network has the capability of learning and adapting to new classes, while refining the existing classes quickly [7]. The training time is short as FMM only needs one pass to learn and refine its decision boundaries. Owing to these advantages, we use FMM to form a hybrid learning model with CART, known as FMM-CART.

Figure 1 depicts a flowchart of the proposed FMM-CART model. Data samples for FMM training are taken from the outputs of Power Spectral Density (PSD). The centroid of the FMM hyperboxes form a crisp data set for tree building based on the CART procedure. The resulting tree is pruned in order to get a simpler tree. Tree optimization is accomplished in order to ensure that there is no over-fitting. The resulting tree is then used for fault classification. The dynamics of FMM-CART are further explained, as follows.
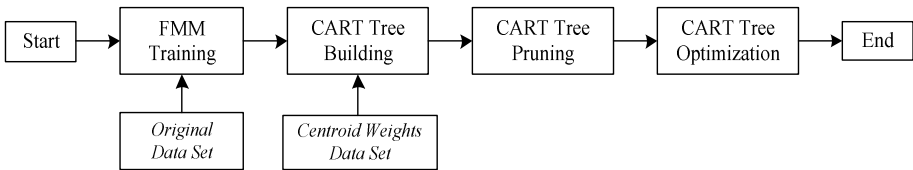
Fig. 1. Flow chart of FMM-CART hybrid model

FMM constitutes the first part of FMM-CART. The FMM classification model is formed using hyperbox fuzzy sets. The size of a hyperbox is controlled by θ, which varies between 0 and 1. When θ increases from a small to a large value, the number of hyperboxes created is reduced. The membership function is set with respect to the minimum and maximum points of a hyperbox, and to the extent to which a pattern fits in the hyperbox. Further details of FMM can be found in [7].

In FMM, the weights contain the minimum and maximum points of a hyperbox. These weights do not represent the centroid of data samples contained in each hyperbox. Anew set of weights that encode the centroid of data samples is introduced in FMM-CART. The centroids of data samples are needed as the learning set of

CART [9] for rule extraction and fault classification.   The centroid is calculated based on a recursive mean computation of data samples contained in each hyperbox. Figure 2 shows a two-dimensional hyperbox with the minimum and maximum points as well as the centroid of data samples contained in the hyperbox.
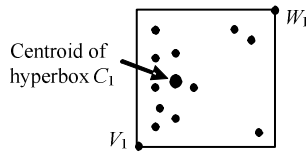
Fig. 2. The centroid point of a two-dimensional hyperbox s

CART has the advantages of handling both numerical and categorical variables that are highly skewed. The structure of the tree does not change when any variable in its logarithm or root square values are replaced. The procedure of CART consists of three basic steps.   The first step is tree building. A tree is built using recursive splitting of nodes, which the rule performs splitting of learning samples into smaller parts.   Based on the decision matrix from the distribution of classes in the learning set, each resulting node is assigned a predicted class. Trees in CART consist of internal nodes and leaf nodes [10]. Tree pruning, the second step, is a sequence of generation of simpler trees. Pruning the tree is necessary to improve the classification accuracy rate. Here, the method of cost-complexity pruning [10] is used.   It is based on a complexity parameter which is gradually increased during the pruning process.   The final step consists of optimal tree selection. The data samples are divided into two sub-sets, one for learning which is used to split nodes and another for testing, which compares the misclassification rates for the sub-trees. As trees grow larger, the misclassification cost for the learning data decreases gradually. The final tree is then be used for motor fault classification.   Further details of CART can be found in [9].

3   Experimental Study

MCSA relies on the spectral analysis of the stator currents, or more precisely the supply currents of an induction motor, to detect an incipient fault.   In this study, PSD is used to extract information from a signal and to describe the power distribution in the frequency domain.   PSD is the Fourier transform of the auto-correlation function of a signal when the signal is stationary [11]. PSD is not restricted to using one specific harmonic for fault detection.   The features extracted from PSD are fed to FMM-CART for fault detection and diagnosis.   An overview of the experimental procedure is shown in Figure 3.
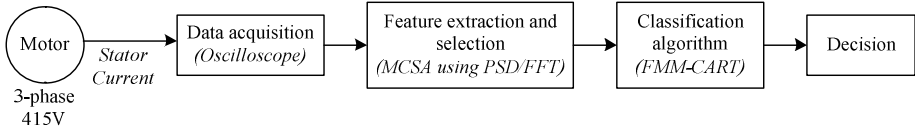
Fig. 3. The experimental procedure of induction motor fault detection and diagnosis

The experiment was conductedwith a laboratory-scale test rig. As shown in Figure 4, the test rig comprises an oscilloscope (1), three current probes (2), an induction motor (3), a belt, shaft, a load inducer (4), and a load controller (5). During the experiment, induction motors with 2 Hp, 415V, 50 Hz, 4-pole were operated with 25%, 50%, 75%, and full load conditions. Three AC current probes were used to measure the stator currents, and the maximum measurement frequency was 50 kHz. Such induction motor configuration is commonly found in many industrial settings[2-3].   As an example, motors smaller than 10 Hp have a market share of 59.6% in Europe [2], and 4-pole motors account for 70% of the total induction motors in Europe [3].
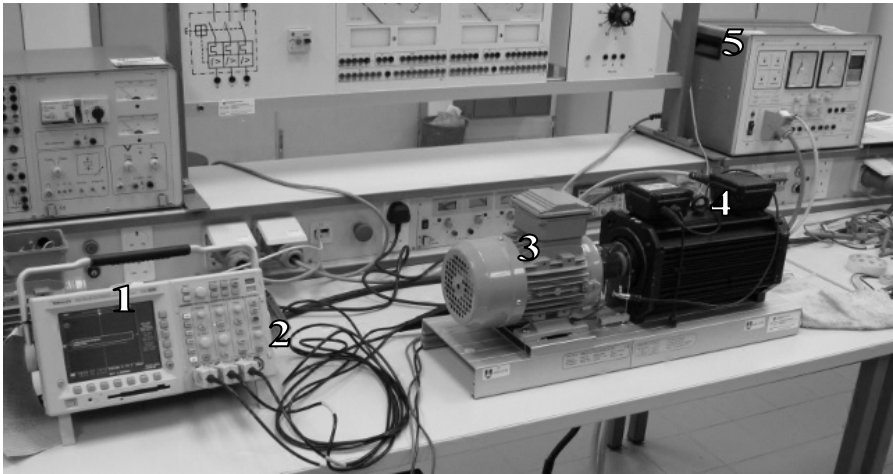
Fig. 4. Experimental setup

Three major types of induction motors faults are evaluated in this study, i.e., stator winding faults, eccentricity problems, and unbalanced voltage conditions. Stator winding faults constitute almost 38% of induction motors faults [4, 12]. The main cause of stator winding faults is short circuit between a phase winding and the ground or between two phases [13].   Note that initial undetected turn-to-turn faults may gradually develop to a major short circuit, and may have a destructive effect on the stator coils [13].   On the other hand, rotor eccentricity-related faults commonly occur as a result of bearing faults [14], which accounts for 40% of induction motor problems [4, 12].   Note that eccentricity faults can cause rotor-to-stator rub, resulting in damage of rotor and/or stator winding.   Another important fault is related to

unbalanced voltage conditions, which occur when the magnitudes of two line voltages become different. Unbalanced voltages lead to a shorterlife span of induction motors.

In this study, the induction motor was operated with 10% turn shorts in one out of the three phases having stator windings short. For eccentricity problems, the induction motors were operated with 30% dynamic eccentricity and 10% static eccentricity. In unbalanced voltage conditions, the induction motor was operated by biasing one of three-phase unbalanced voltages by 5% (395V), followed by 10% (374V). The PSD outputs comprised a 1000 Hz frequency spectrum, from the 1st to 19th harmonics. Note that in a balanced three-phase system, the triplen harmonic voltages are zero [15]. As such, the 3rd, 9th, and 15th harmonics were discarded in this study. The FMM-CART model received a total of 21 features comprising the 1st, 5th, 7th, 11th, 13th, 17th, and 19th harmonics from phases A, B, and C. The output of FMM-CART was the prediction of four motor conditions, with classes 1, 2, 3, 4 indicating healthy motor, stator winding faults, eccentricity faults, and unbalanced voltage conditions, respectively.

3.1   Experimental Results and Analysis

Multiple trial-and-error runs were conducted using current signals from the motor with the parameters as per the specifications in Section 3. All the data samples were normalized between 0 and 1. The cross validation method, a realistic assessment of the overall network performance, was adopted. A total of 1000 data samples were generated, and divided equally into five sub-sets. Four sub-sets were used for learning and the remaining for testing. As such, the experiment was repeated five times, each time with a new sub-set of data samples for testing. The results (averages and standard deviations) were computed using the bootstrap method [16] with 5000 re-samplings. Using an Intel Core 2 Duo 2.80 GHz processor with 4 GB RAM on MATLAB® R2010a, the computational time taken by a single run of the cross-validation experiment was also recorded. For performance comparison, four different models, viz., the Multi-Layer Perceptron (MLP) network, FMM, CART, and FMM-CART were evaluated. Table 1 shows a summary the overall results.

Table 1. Classification results ofinduction motor conditions

Model	Accuracy (%)	Standard Deviation of Accuracy (%)	Model Complexity	Time (sec)
MLP	94.77	12.76	19 Hidden Nodes	2.01
FMM	96.61	3.21	8Hyperboxes	0.52
CART	98.48	2.12	4 Leafs	0.88
FMM-CART	99.08	1.05	4 Leafs	0.83

From Table 1, MLP produced the lowest accuracy of 94.77%, with 19 hidden nodes. FMM yielded 96.61% accuracy with 8hyperboxes. CART and FMM-CART achieved 98.48% and 99.08% accuracy, each with 4 leafs. Note that CART used all data samples to produce 98.48% accuracy, while FMM used only 30% of data samples to achieve 96.61% accuracy. In FMM-CART, the hyperboxes resulted from FMM training further reduced the number of data samples to only 15%. These data samples were used by FMM-CART to produce the best accuracy rate of 99.08%.

Notice that the standard deviation of accuracy from FMM-CART is the lowest, which indicates the stability of its performances. Nevertheless, the computational time taken by FMM-CART is longer than that of FMM, but is slightly shorter than that of CART. The decision tree produced by FMM-CART is the same as that of CART, i.e. with four leafs which is less complex as compared with those from MLP and FMM. In other words, FMM is useful for extracting the important features for classification by using CART. Figure 5 depicts the decision tree resulted from FMM-CART. Analysis of the decision tree is as follows.
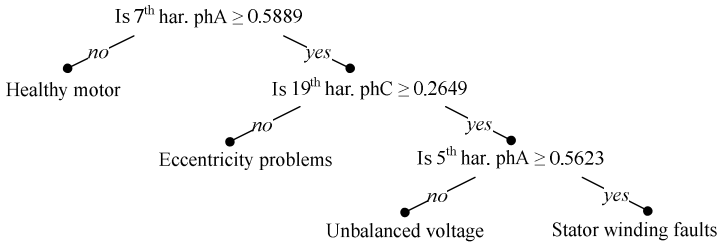
Fig. 5. The decision tree for all motor conditions

The tree starts by splitting the 7th harmonic of phase A. For values below 0.5889, FMM-CART classifies the input as a healthy motor. For values above 0.5889, the tree splitsat the 19th harmonic phase of C. For values below 0.2649, FMM-CART classifies the input as an eccentricity problem. The tree splits again for values above 0.2649. For values less than 0.5623 forthe 5th harmonic of phase A, FMM-CART classifies the input as an unbalanced voltage condition. Otherwise, the input belongs to a stator winding fault.These rules are in agreement with findings reported in the literature. Lee et al [17] stated that the 5th harmonic is the dominant harmonic in unbalanced voltages. Nandi et al [18] stated that the 19th harmonic has a stronger energy content which is good for monitoring eccentricity faults while Briz et al [19] stated that the 5th harmonic was the best candidate in detecting stator winding faults.

4   Conclusion

A hybrid FMM-CART model for induction motor fault detection and diagnosis has beendescribed. In the proposed approach, MCSA has been used for stator current signal acquisition while PSD has been used for converting the current signals into their frequency spectra. The resulting motor current harmonics form the input features to FMM-CART for fault detection. An accuracy of 99.08% has been achieved by FMM-CART, which outperforms MLP, FMM, and CART. The decision tree from FMM-CART has also shown to be useful in analyzing induction motor fault conditions. In short, FMM-CART is able to make rapid and accurate predictions for fault detection and diagnosis of induction motors.

References

1. Cusidó, J., Romeral, L., Ortega, J.A., Rosero, J.A., Espinosa, A.G.: Fault Detection in Induction Machines Using Power Spectral Density in Wavelet Decomposition. IEEE Trans. Industrial Electronics 55, 633–643 (2008)
2. Almeida, A.T.: Energy Using Product Directive Preparatory Study, Lot 11: Motors, Analysis of Existing Technical and Market Information. DG TREN (2006)
3. Commission of the European Communities, Full Impact Assessment, C(2009) 5675, SEC(2009) 1014 (2009)
4. Rodríguez, P.V.J., Negrea, M., Arkkioa, A.: A Simplified Scheme for Induction Motor Condition Monitoring. Mechanical System and Signal Processing 22, 1216–1236 (2008)
5. Tran, V.T., Yang, B.S., Oh, M.S., Tan, A.C.C.: Fault Diagnosis of Induction Motor Based on Decision Trees and Adaptive Neuro-Fuzzy Inference. Expert Systems with Applications 36, 1840–1849 (2009)
6. Palmero, G.I.S., Santamaria, J.J., de la Torrea, E.J.M., González, J.R.P.: Fault Detection and Fuzzy Rule Extraction in AC Motors by a Neuro-Fuzzy ART-Based System. Engineering Applications of Artificial Intelligence 18, 867–874 (2005)
7. Simpson, P.K.: Fuzzy Min-Max Neural Networks - Part 1: Classification. IEEE Trans. Neural Networks 3, 776–786 (1992)
8. Simpson, P.K.: Fuzzy Min-Max Neural Networks - Part 2: Clustering. IEEE Trans Fuzzy Systems 1, 32–45 (1993)
9. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: Classification and Regression Trees. Chapman and Hall, Belmont (1984)
10. Lewis, R. J.: An Introduction to Classification and Regression Tree (CART) Analysis. Society for Academic Emergency Medicine (2000)
11. Herman, P., Prasad, G., McGinnity, T.M., Coyle, D.: Comparative Analysis of Spectral Approaches to Feature Extraction for EEG-Based Motor Imagery Classification. IEEE Trans. Neural Systems and Rehabilitation Engineering 16, 317–326 (2008)
12. IEEE Motor Reliability Working Group, Report of Large Motor Reliability Survey of Industrial and Commercial Installations, Part I. IEEE Trans. Industrial Applications 21, 853–864 (1985)
13. Awadallah, M.A., Morcos, M.M.: Application of AI Tools in Fault Diagnosis of Electrical Machines and Drives-An Overview. IEEE Trans. Energy Conversion 18, 245–251 (2003)
14. Rosero, J.A., Cusido, J., Garcia, A., Ortega, J.A., Romeral, L.: Broken Bearings and Eccentricity Fault Detection for a Permanent Magnet Synchronous Motor. In: IECON 2006 32nd Annual Conference on IEEE Industrial Electronics, pp. 964–969. IEEE Press, New York (2006)
15. Pedra, J., Sainz, L., Córcoles, F.: Harmonic modeling of induction motors. Electric Power Systems Research 76, 936–944 (2006)
16. Efron, B.: Bootstrap Methods: Another Look at the Jackknife. Annals of Statistics 7, 1–26 (1979)
17. Lee, G.M., Lee, D.C., Seok, J.K.: Control of Series Active Power Filters Compensating for Source Voltage Unbalance and Current Harmonics. IEEE Trans. Industrial Electronics 51, 132–139 (2004)
18. Nandi, S., Ilamparithi, T.C., Lee, S.B., Hyun, D.: Detection of Eccentricity Faults in Induction Machines Based on Nameplate Parameters. IEEE Trans. Industrial Electronics 58, 1673–1683 (2011)
19. Briz, F., Degner, M.W., Guerrero, J.M., García, P.: Stator Windings Fault Diagnostics of Induction Machines Operated From Inverters and Soft-Starters Using High-Frequency Negative-Sequence Currents. IEEE Trans. Industrial Applications 45, 1637–1646 (2009)

A Multimodal Information Collector
for Content-Based Image Retrieval System

He Zhang, Mats Sjöberg, Jorma Laaksonen, and Erkki Oja

Department of Information and Computer Science,
Aalto University School of Science, Espoo, Finland
{he.zhang,mats.sjoberg,erkki.oja,jorma.laaksonen}@aalto.fi

Abstract. Explicit relevance feedback requires the user to explicitly refine the search queries for content-based image retrieval. This may become laborious or even impossible due to the ever-increasing volume of digital databases. We present a multimodal information collector that can unobtrusively record and asynchronously transmit the user's *implicit* relevance feedback on a displayed image to the remote CBIR server for assisting in retrieving relevant images. The modalities of user interaction include eye movements, pointer tracks and clicks, keyboard strokes, and audio including speech. The client-side information collector has been implemented as a browser extension using the JavaScript programming language and has been integrated with an existing CBIR server. We verify its functionality by evaluating the performance of the gaze-enhanced CBIR system in on-line image tagging tasks.

Keywords: Implicit relevance feedback, JavaScript, gaze tracking, content-based image retrieval, image tagging.

1 Introduction

Relevance feedback has been widely utilized in content-based image retrieval (see [1] for an extensive survey). Often, people need to attentively indicate or answer whether or not the retrieved information is relevant, and thus give *explicit relevance feedback*. With large databases and long retrieval sessions, this will inevitably become a laborious task. The interest for using *implicit relevance feedback* [2], although less accurate than explicit, has increased in recent years. By using implicit relevance feedback, an information retrieval system can unobtrusively record the user's behavior, such as gaze direction, facial expressions, body gestures etc., and use this information to infer the user's search preferences [3]. Moreover, a combination of explicit and implicit feedback can even better model the user's potential interests [4].

In the current work, we present a multimodal information collector that can unobtrusively record and transmit the user's implicit feedback, in addition to the explicit feedback, to the remote CBIR server for image retrieval. The feedback modalities include eye movements, pointer tracks and clicks, keyboard strokes, and audio including but not limited to speech. We focus on using gaze as a primary feedback modality since eye movements have earlier been found to have

a strong correlation with human cognitive processes (see [5] for a thorough review). Using eye movements as an implicit feedback source is a relatively new research area. However, eye movements have already demonstrated strong potentials in inferring people's interests in tasks such as image retrieval [6] and image ranking [7].

A similar information collector can be found in [8], where a prototype attentive information system was implemented to track the user's behavior and suggest helpful information to the user. However their system was not evaluated quantitatively. In [9], a Web-Accessible Multimodal Interfaces (WAMI) toolkit was developed at MIT. Their approach was tightly connected to the use of speech input and audio output in web applications and had no directions to eye movement analysis.

The following section describes the implementation and operation principles of the multimodal information collector. Section 3 introduces the gaze-enhanced CBIR system. In Section 4, we verify the functionality of the information collector by evaluating the system performance with real user experiments. Section 5 concludes the paper and discusses our future work.

2   Implementation and Operation Principles

Figure 1 illustrates the overall schematic diagram of a client-server based image retrieval system with four forms of user interaction modalities collected at the client side. The multimodal information collector has been implemented as an extension of Mozilla Firefox, which is a free and open source web browser of great popularity today.

2.1   Client Implementation

The client-side collector is programmed by using the JavaScript language since it is the primary implementation language of Mozilla Firefox extensions and supports prototype-based object construction and object-oriented programming including class inheritance.
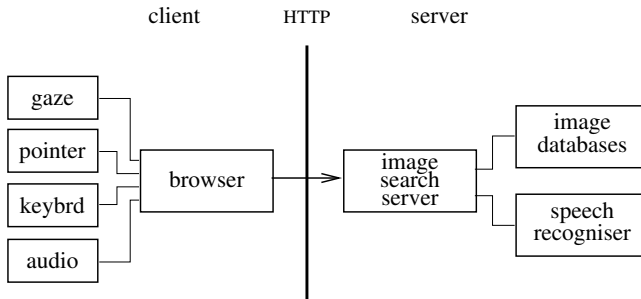
Fig. 1. The block diagram of our system capable of transferring multimodal feedback from a browser client to a content-based image search server
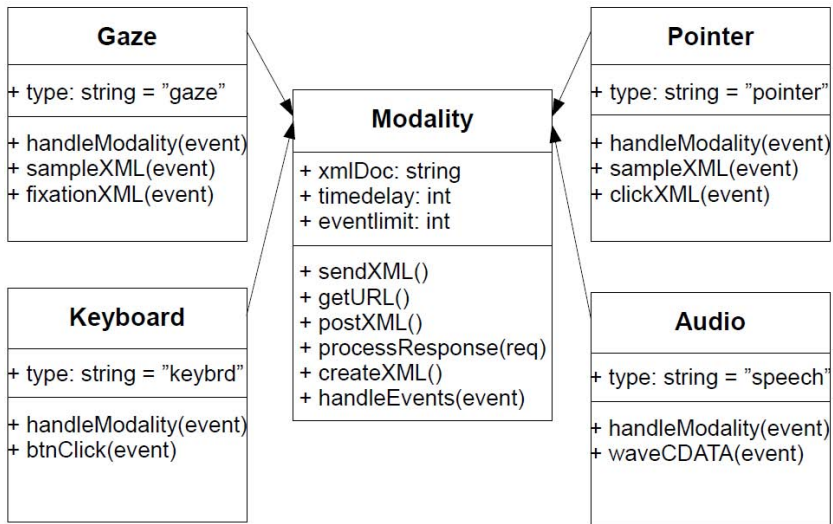
Fig. 2. A class diagram showing the relationships between the **Modality** super-class and its four sub-classes, **Gaze**, **Pointer**, **Keyboard**, and **Audio**

Figure 2 shows a class inheritance diagram of the JavaScript implementation. The roles of the classes are defined as follows:

Modality is defined to be the super-class of the classes for modalities of user interaction. These all collect the input and store them internally in the `xmlDoc` object which is then periodically transmitted to the server in XML-formatted packets.

Gaze is defined to be a sub-class of Modality. This module detects the user's gaze samples and fixations on a displayed image and converts them into 2D coordinates relative to the shown image.

Pointer is defined to be a sub-class of Modality. This module detects the user's pointer movements (and clicks) and converts them into 2D coordinates.

Audio is defined to be a sub-class of Modality. This module detects and converts the user's voice into binary audio data for the speech recognizer.

Keyboard is defined to be a sub-class of Modality. This module detects and records the user's keyboard events (strokes).

Each sub-class inherits all the functions defined in **Modality**, but adds its own methods for that particular functionality. The method `handleModality()` is overridden in each sub-class for a specific modality. It generates the XML data structure for that particular interaction modality. For example, in the class of **Gaze**, the method `handleModality()` calls the `sampleXML()` method to generate XML data for gaze samples, and `fixationXML()` for gaze fixations. Then the general method `handleEvents()` in the super-class periodically sends the collected data to the specified server URL. Similar routines apply to the other three sub-classes.

2.2   Client-Server Interaction

The client-server communications are based on the World Wide Web Consortium
(W3C) XMLHttpRequest protocol[1], which has recently been employed exten-
sively for implementing *Asynchronous JavaScript and XML* (AJAX)[2] type of
asynchronous content updates in web applications.
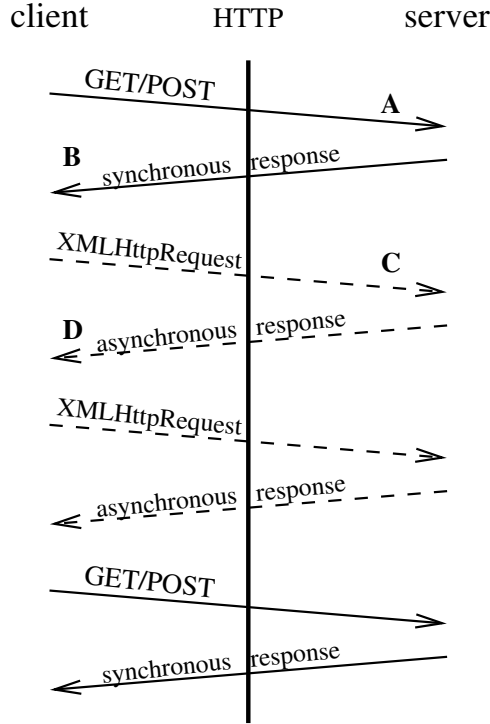
Fig. 3. The message exchange diagram of synchronous and asynchronous communica-
tions between the image search client and the content-based image retrieval server

Figure 3 shows the messaging diagram of synchronous and asynchronous com-
munications between an image database server and a browser client. The image
retrieval session is initiated by the client requesting the server to present some
visual content for inspection. The server returns with a conventional synchronous
HTTP action a set of images to the browser, which then presents them to the
user. Together with the XML/HTML page containing the images, the server can
also specify an URL from its own URL space where the client can send asyn-
chronous user interaction data. To our knowledge, this is a novel idea not used
in any existing content-based image retrieval systems.

The message types transferred between the server and the client have been identified as **A**, **B**, **C** and **D** in Figure 3. Their exact roles in the communication of multimodal relevance feedback are omitted here due to the space limit.

3   Gaze-Enhanced Content-Based Image Retrieval

3.1   PicSOM CBIR System

We have integrated the multimodal information collector with an existing CBIR server named PicSOM[3] [10], which is a content-based image retrieval system developed since 1998, first at the Helsinki University of Technology and then at the Aalto University. PicSOM uses the principles of *query by example* and *relevance feedback* in implementing iterative and interactive image retrieval.

The unique approach used in PicSOM is to have several Self-Organizing Maps (SOMs) [11] in parallel to index and determine the similarity of images. These parallel SOMs have been trained with separate data sets obtained by using different feature extraction algorithms on the same objects. The extracted image features [12] include RGB histogram, DCT coefficients, edge statistics etc.

As the SOM maps visually similar images near to each other, this motivates to spread the relevance feedback given for the viewed images to their neighboring images on the map surface. Images marked as relevant are first given positive and those marked as non-relevant are given negative values on the map surface. These relevance values are then smoothed and spread around with low-pass filtering. Images with the largest resulting relevance scores are then shown to the user.

3.2   Using Gaze Patterns as Implicit Relevance Feedback

Based on the received gaze coordinates at the server, we calculate for each viewed image a 19-dimensional feature vector as specified in Table 1. These features have been used in image retrieval and ranking tasks before [6,7]. The relevance predictions for the viewed images are obtained with a simple logistic regression model created with separate training data.

In the PicSOM system, the gaze-based implicit relevance estimates are combined with the click-based explicit relevance feedback values. In this process the gaze-based regressor outputs are always in the range of $[0, 1]$ and the larger the value, the more probably the image is relevant. These values are then summed with the $+1$ and -1 values given for the clicked and non-clicked images, respectively. The combined relevance values are finally placed in the SOM units and spread to their neighbors with low-pass filtering similarly to PicSOM's normal operation.

3.3   Automatic Speech Recognition

The speech recognition system used in the experiments has been developed by the speech group in the Department of Information and Computer Science at

---

[3] http://www.cis.hut.fi/picsom

Table 1. Eye movement features collected1 at the client side

Number	Name	Description
1	numMeasurements	log of total time of viewing the image
2	numOutsideFix	total time for measurements outside fixations
3	ratioInsideOutside	percentage of measurements inside/outside fixations
4	speed	average distance between two consecutive measurements
5	coverage	number of subimages covered by measurements[1]
6	normCoverage	coverage normalized by numMeasurements
7	pupil	maximal pupil diameter during viewing
8	nJumps1	number of breaks[2] longer than 60ms
9	nJumps2	number of breaks[2] longer than 600ms
10	numFix	total number of fixations
11	meanFixLen	mean length of fixations
12	totalFixLen	total length of fixations
13	fixPrct	percentage of time spent in fixations
14	nJumpsFix	number of re-visits (regressions) to the image
15	maxAngle	maximal angle between two consecutive saccades[3]
16	firstFixLen	length of the first fixation
17	firstFixNum	number of fixations during the first visit
18	distPrev	distance to the fixation before the first visit
19	durPrev	duration of the fixation before the first visit

[1]The image was divided into a regular grid of 4×4 subimages, and covering a subimage means that at least one measurement falls within it. [2]A sequence of measurements outside the image occurring between two consecutive measurements within the image. [3]A transition from one fixation to another.

Aalto University. The speech signal is sampled using 16 kHz sampling rate and 16 bits. The signal is then represented with 12 MFCC (mel-frequency cepstral coefficients) and the log-energy along with their first and second differentials. Above features are calculated in 16 ms windows with 8 ms overlap. Cepstral mean subtraction (CMS) and a maximum likelihood linear transformation, which is estimated in training, are applied to the features. For the acoustic model, we use state-clustered Hidden Markov triphone models that have 5062 states modeled with 32 Gaussians.

4   Experiments

In this section, we verify the functionality of the proposed multimodal information collector, and evaluate the gaze-enhanced CBIR system in an image tagging scenario. Automatic tagging is a useful but still not fully reliable means for associating keyword-type information to unannotated images. In the current state of the art, human effort is still needed for checking and correcting the tags [13]. The tag correction process can be seen as a special case of content-based image retrieval where the goal is to quickly correct the erroneously-tagged images.

4.1   CBIR-Assisted Image Tag Correction

Let us consider a CBIR setup where the viewed images are such that an automatic image annotation system has assigned all of them some particular tag or keyword based on their visual properties. The considered images are thus visually quite similar to each other, but due to imperfections in the assignment, there are bound to be semantic differences or tagging errors among them. The burden of a user who needs to check and correct the automatically assigned tags would be eased if the wrongly-tagged images could be found as early as possible.

This can be understood as a complementary setting for the conventional interactive CBIR setting. Now the relevant images are not those that resemble the target image, but those that are semantically different from the other, correctly-tagged ones. Nevertheless, CBIR techniques can be used to speed up retrieving of such images. This time, the search will be driven more by the negative relevance feedback, given to the correctly-tagged images. The system will then retrieve more and more images that are different from the typical correctly-tagged images and are thus more likely to be the wrongly-tagged ones.

4.2   Setups and Evaluations

Data. We used the *train* subset of the PASCAL Visual Object Classes Challenge 2007 (VOC2007) data set [14] with a total of 2501 annotated images that cover 20 overlapping categories. To ease the burden of users, we randomly selected 16 categories and divided them into two groups:

1. correctly-tagged: *car, dog, bicycle, person, motorbike, train*
2. wrongly-tagged: *sheep, horse, aeroplane, boat, bus, bottle, dining table, potted plant, sofa, tv-monitor*

Experiment Setup. We recruited 18 test subjects both males and females from several departments at the Aalto University. The mean age of the test subjects was 27.2 years old, ranging from 23 to 34 with good balances in between. Very few of the users had experiences in image tagging and only one user had experiences in gaze tracking.

Each subject was asked to perform six tagging tasks. For each task, the user had to check and correct the tags of one particular category from group 1. Before each task, the system randomly selected 40 images of that category and another 40 images of the ten categories from the wrongly-tagged group. Thus half of the images were always tagged correctly. During each task, the system showed a total of 40 images, contained in five image pages each having eight images. After each task, the user was asked of his or her subjective opinions whether the corresponding variant facilitated the tagging task, and whether it was reliable and fast enough.

Feedback Modalities. The following relevance feedback modality types or variants of the system were compared:

1. *Baseline*: The user corrects the image tag by selecting the corresponding category name from the drop-down menu under the image. No CBIR or speech recognition techniques are used.

2. *Explicit*: The user clicks the pointer over the wrongly-tagged image and speaks the desired category name into the microphone. Only explicit relevance feedback from pointer clicks are used.
3. *Implicit*: The tag correction is similar as in *explicit*. However, the user's eye movements are unobtrusively recorded by a Tobii eye tracker[4]. Both explicit pointer relevance and implicit gaze relevance feedback are used.

For the baseline variant all the 40 images presented to the user were randomly chosen, whereas for the other two variants only the eight images in the first page were random while the images in the remaining four pages were selected by the relevance feedback information.

The Evaluation and Results of Image Retrieval. The measure of performance is the number of images that the user corrects in one tagging task, which gives reflection on how well the system retrieves wrongly-tagged images. Table 2a gives the quantitative performance of the three variants for each user. Although the relative performance of the variants varies between users, it is clear that explicit and implicit feedback are better than the baseline. This can be seen from

Table 2. (a) The rounded average numbers of images that each user corrected when using the three variants of the system. The best performance(s) are marked in bold for each user. (b) The rounded average numbers of images corrected for each category averaged over 18 users. (c) Means and variances over the 18 users for the three system variants.

(a)

User	Baseline	Explicit	Implicit
1	16	23	**24**
2	**22**	21	**22**
3	**26**	25	**26**
4	19	**27**	**27**
5	24	**27**	23
6	23	**27**	26
7	25	25	**26**
8	23	**29**	19
9	23	**26**	24
10	18	24	**28**
11	15	**26**	22
12	25	22	**26**
13	22	**28**	27
14	22	**28**	24
15	**27**	25	22
16	28	**29**	27
17	27	**28**	25
18	26	24	**27**

(b)

Category	Baseline	Explicit	Implicit
car	20	**23**	22
dog	22	**28**	26
bicycle	23	**27**	25
person	23	26	**31**
motorbike	**26**	24	23
train	22	**26**	23

(c)

	Baseline	Explicit	Implicit
mean	22.83	25.78	24.72
var	14.15	5.48	5.74

---

[4] http://www.tobii.com/

the averages, and from the fact that the baseline has the best performance in only three cases. For users 10 and 12, the implicit variant of the system retrieved about 17% more of the wrongly-tagged images than the explicit variant.

Table 2b gives the quantitative performance for each tagging category averaged over all the users. Similarly, the performances of the explicit and implicit variants are better than that of the baseline type, except for the *motorbike* category. The reason is probably because of the overlapping categories of the images in the VOC2007 database. For example, an image tagged as *motorbike* usually contains a person riding on it, which might cause users to tag it as *person*. However, for the *person* category, the implicit variant of the system retrieved about 20% more of the wrongly-tagged images than the explicit variant did.

The Evaluation of User Experience. A close examination of the qualitative feedback from the users (questionaires) indicates that most of the test subjects (between 66% and 75%) believed that all the variants help to facilitate the tagging tasks, though they had to spend extra efforts in adapting to the eye tracker and microphone. As for reliability, about 82% of the test subjects considered the explicit variant with speech input to be the most reliable one, whereas respectively 56% and 50% of the subjects marked the implicit variant and baseline variant to be reliable. As for speed, the implicit variant with gaze tracking received the highest vote of 64%, followed by the explicit variant of 56% and the baseline variant of 43%.

5   Conclusions and Future Work

We have developed a novel client-side multimodal information collector that has been implemented as a Firefox browser extension for asynchronously transmitting versatile user interaction modalities, such as eye movements on the displayed images, to the remote CBIR server. The collector has been integrated with an existing neural-network-based CBIR system that is made capable of handling XMLHttpRequest messages from the client.

We have verified the functionality of the collector by evaluating the performance of the gaze-enhanced CBIR system with real user image tagging tasks. The quantitative results showed that both the explicit variant using pointer clicks and the implicit variant using gaze tracking patterns can to some extent speed up the search and correction of wrongly-tagged images, compared to the baseline variant with drop-down menus. The qualitative results revealed that the implicit variant enhanced by gaze and speech was believed to have the highest speed among the three.

Our next step is to improve the client-server system by fusing the user's eye movements and mouse tracks with more users involved. This time the image database will be expanded to contain millions of images sampled from Flickr and ImageNet.

References

1. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: Ideas, influences, and trends of the new age. ACM Computing Surveys 40(2), 1–60 (2008)
2. Kelly, D., Teevan, J.: Implicit feedback for inferring user preference: a bibliography. SIGIR Forum 37(2), 18–28 (2003)
3. Zhang, H., Koskela, M., Laaksonen, J.: Report on forms of enriched relevance feedback. Technical Report TKK-ICS-R10, Helsinki University of Technology (2008)
4. Hardoon, D.R., Shawe-Taylor, J., Ajanki, A., Puolamäki, K., Kaski, S.: Information retrieval by inferring implicit queries from eye movements. In: Eleventh International Conference on Artificial Intelligence and Statistics (2007)
5. Rayner, K.: Eye movements in reading and information processing: 20 years of research. Psychological Bulletin 124(3), 372–422 (1998)
6. Klami, A., Saunders, C., de Campos, T., Kaski, S.: Can relevance of images be inferred from eye movements? In: Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval, pp. 134–140. ACM (2008)
7. Hardoon, D., Pasupa, K.: Image ranking with implicit feedback from eye movements. In: Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications, pp. 291–298. ACM (2010)
8. Maglio, P.P., Campbell, C.S.: Attentive agents. Commun. ACM 46(3), 47–51 (2003)
9. Gruenstein, A., McGraw, I., Badr, I.: The WAMI Toolkit for developing, deploying, and evaluating web-accessible multimodal interfaces. In: Proceedings of Tenth International Conference on Multimodal Interfaces (ICMI 2008), Chania, Greece (October 2008)
10. Laaksonen, J., Koskela, M., Oja, E.: PicSOM—Self-organizing image retrieval with MPEG-7 content descriptions. IEEE Transactions on Neural Networks, Special Issue on Intelligent Multimedia Processing 13(4), 841–853 (2002)
11. Kohonen, T.: Self-Organizing Maps, 3rd edn. Springer Series in Information Sciences, vol. 30. Springer, Berlin (2001)
12. Viitaniemi, V., Laaksonen, J.: Evaluating the performance in automatic image annotation: example case by adaptive fusion of global image features. Signal Processing: Image Communications 22(6), 557–568 (2007)
13. Ames, M., Naaman, M.: Why we tag: motivations for annotation in mobile and online media. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 971–980. ACM, New York (2007)
14. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2007, VOC 2007 (2007)

Graphical Lasso Quadratic Discriminant Function for Character Recognition

Bo Xu[1], Kaizhu Huang[1], Irwin King[2,4], Cheng-Lin Liu[1],
Jun Sun[3], and Naoi Satoshi[3]

[1] National Laboratory of Pattern Recognition, Institute of Automation of Chinese
Academy of Sciences, Beijing 100190, P.R. China
[2] Dept. of CSE, The Chinese University of Hong Kong
[3] Fujitsu Research and Development Center, Beijing, China
[4] AT&T Labs Research, San Francisco, USA
{box,kzhuang,liucl}@nlpr.ia.ac.cn, king@cse.cuhk.edu.hk,
{sunjun,naoi}@cn.fujitsu.com

Abstract. The quadratic discriminant function (QDF) derived from the multivariate Gaussian distribution is effective for classification in many pattern recognition tasks. In particular, a variant of QDF, called MQDF, has achieved great success and is widely recognized as the state-of-the-art method in character recognition. However, when the number of training samples is small, covariance estimation involved in QDF will usually be ill-posed, and it leads to the loss of the classification accuracy. To attack this problem, in this paper, we engage the graphical lasso method to estimate the covariance and propose a new classification method called the Graphical Lasso Quadratic Discriminant Function (GLQDF). By exploiting a coordinate descent procedure for the lasso, GLQDF can estimate the covariance matrix (and its inverse) more precisely. Experimental results demonstrate that the proposed method can perform better than the competitive methods on two artificial and six real data sets (including both benchmark digit and Chinese character data).

Keywords: Graphical Lasso, Quadratic Discriminant Function, Character Recognition.

1 Introduction

In many pattern recognition tasks, it is very common to assume that the data follow a Gaussian distribution. The quadratic discriminant function (QDF) derived from the multivariate Gaussian distribution can then be used for classification. Despite of its simplicity, QDF or its variants have achieved great success in many fields. The parameters involved in QDF, e.g., the mean and the covariance, are often obtained via the principle of the maximization-likelihood Estimation (MLE) [6]. MLE has a number of attractive features. First, it usually has good convergence properties as the number of training samples increases. Furthermore, it can often be simpler than alternative methods, such as Bayesian techniques.

However, when the number of training samples is small (especially when compared to dimensionality), the estimated covariance based on MLE could be often ill-posed, making the covariance matrix singular; this further leads its inverse matrix cannot be computed reliably.

To solve this problem, there have been a number of approaches in the literature. Modified Quadratic Discriminant Function (MQDF) [8] is proposed to replace the minor eigenvalues of covariance matrix of each class with a constant parameter. This small change proves very effective and has made MQDF a state-of-the-art classifier in character recognition. However, the substitution of minor eigenvalues with a constant inevitably loses some class information. Meanwhile, the cutoff threshold of minor eigenvalues and the constant selection are critical for the final performance. Liu et al. [11] proposed a discriminative learning algorithm called Discriminative Learning QDF (DLQDF). It optimizes the parameters of MQDF with the aim to improve the classification accuracy based on the criterion of Minimum Classification Estimation (MCE). Similar to MQDF, DLQDF has the same problem in parameter selection. Alternatively, the Regularized Discriminant Analysis (RDA) [5] improves the performance of QDF by covariance matrix interpolation. Hoffbeck and Landgrebe further extended RDA by optimizing the interpolation coefficients [7]. Empirical results showed that these two algorithms can usually improve the classification performance of QDF. However, the improvements are also dependent on two critical parameters β and γ. In short, all of the above-mentioned methods need empirical settings of parameters to achieve the best results, which are however both time-consuming and task-dependent in real applications.

Different from the above approaches, in this paper, we present a novel method, called the Graphical Lasso Quadratic Discriminant Function (GLQDF). By engaging the graphical lasso, the covariance estimation of the ordinal QDF can be successfully conducted even when the number of training samples is very small. Moreover, we can estimate the inverse of the covariance directly and hence avoid singular problems involved in QDF. One appealing feature is that the whole process is parameter-insensitive. This presents one big advantage over the other methods.

The rest of the paper is organized as follows. In the next section, we make an overview of QDF and MQDF. In Section 3, we introduce our novel GLQDF in details. In Section 4, we conduct a series of experiments to verify our method. Finally, we set out concluding remarks in Section 5.

2   Review of QDF and MQDF

2.1   Quadratic Discriminant Function

Let d be the dimension of the feature. The probability density function of d-dimensional normal distribution is:

$$p(x) = \frac{1}{(2\pi)^{d/2} \left| \Sigma \right|^{1/2}} \exp\{-\frac{1}{2} \left(x - u \right)^t \Sigma^{-1} (x - u)\} \, , \tag{1}$$

where x is a d-component vector, μ is the mean vector, and Σ is the $d \times d$ covariance matrix. The quadratic discriminant function is derived from Eq.(1) as follows:

$$g(x) = (x - \mu)^t \Sigma^{-1}(x - \mu) + \log |\Sigma| = \sum_{i=1}^{d} \frac{((x-\mu)^t \varphi_i)^2}{\lambda_i} + \sum_{i=1}^{d} \log \lambda_i \ , \qquad (2)$$

where λ_i is the i-th eigenvalue of Σ sorted by descending order and φ_i is the eigenvector that corresponds to λ_i. This function will lead to the optimal classifier, provided that (1) the actual distribution is normal, (2) the prior probabilities of all categories are equal and (3) the parameters μ and Σ can be reliably provided. However, since the parameters are usually unknown, the sample mean vector $\hat{\mu}$ and sample covariance matrix $\hat{\Sigma}$ are used.

$$\begin{aligned} \hat{g}(x) &= (x - \hat{\mu})^t \hat{\Sigma}^{-1}(x - \hat{\mu}) + \log \left| \hat{\Sigma} \right| \\ &= \sum_{i=1}^{d} \frac{((x-\hat{\mu})^t \hat{\varphi}_i)^2}{\hat{\lambda}_i} + \sum_{i=1}^{d} \log \hat{\lambda}_i \ . \end{aligned} \qquad (3)$$

Here, λ_i is the i-th eigenvalue of $\hat{\sigma}$ and $\hat{\varphi}_i$ is the eigenvector. It is well-known that small eigenvalues in Eq.(3) are usually inaccurate; this hence causes the reduction of recognition accuracy. Moreover, the computational cost of Eq.(3) is $O(d^3)$ for d-dimensional vectors, which may be computationally difficult when the dimension is high.

2.2 Modified Quadratic Discriminant Function

MQDF is a modified version of the ordinary QDF. QDF suffers from the quadratic number of parameters, which cannot be estimated reliably when the number of samples per class is smaller than the feature dimensionality. MQDF reduces the complexity of QDF by replacing the small eigenvalues of covariance matrix of each class with a constant. Consequently, the small eigenvectors will disappear in the discriminant function. This reduces both the space and the computational complexity. More importantly, this small change proves to improve the classification performance significantly. Denote the input sample by a d-dimensional feature vector $x = (x_1, x_2, x_3, , x_d)^T$. For classification, each class c_i is assumed to have a Gaussian density $p(x|c_i) = N(u_i, \sigma_i)$, where μ_i and σ_i are the class mean and covariance matrix, respectively. Assuming equal a priori class probabilities, the discriminant function is given by the log-likelihood:

$$- 2 \log p\left(x|c_i\right) = (x - \mu_i)^T \Sigma_i^{-1}(x - \mu_i) + \log |\Sigma_i| + CI \qquad (4)$$

where CI is a class-independent term, and is usually omitted. We take the minus log-likelihood to make the discriminant function a distance measure. The covariance matrix Σ_i can be diagonalized as: Λ_i, where $\Lambda_i = diag[\lambda_{i1}, ..., \lambda_{ik}, ..., \lambda_{id}]$ has the eigenvalues of λ_{ik} (in descending order) as diagonal elements, φ_{ik} is an ortho-normal matrix comprising as columns the eigenvectors of λ_{ik}. Replacing the minor eigenvalues with a constant, i.e., replacing Λ_i with $diag[\lambda_{i1}, , \lambda_{ik}, \delta_i, , \delta_i]$

(k is the number of principal eigenvectors to be retained), the discriminant function of Eq. (3) becomes what we call MQDF:

$$
\begin{aligned}
g(x, c_i) = & \sum_{j=1}^{k} \frac{((x-\mu_i)^t \varphi_{ij})^2}{\lambda_{ij}} + \sum_{j=1}^{k} \log \lambda_{ij} \\
& + \frac{1}{\delta_i} \left(\|x - \mu_i\|^2 - \sum_{j=1}^{k} \left|(x - \mu_i)^T \varphi_{ij}\right|^2 \right) + (d - k) \log \delta_i \ ,
\end{aligned}
\tag{5}
$$

where $i, j = 1, \ldots, k$, are the principal eigenvectors of the covariance matrix of class ω_i. In classification, the input pattern is classified to the class of minimum quadratic distance (MQDF), and multiple candidate classes are ordered in ascending order of distances.

3   Graphical Lasso Quadratic Discriminant Function

In recent years, a number of researchers have proposed the estimation of Gaussian models through the use of L_1 (lasso) regularization, which increase the sparsity of the inverse covariance. Meinshausen and Buhlmann [12] took a simple approach to this problem. They estimated a sparse model by fitting a lasso model to each variable while using the others as predictors. Other researchers have proposed algorithms for the exact maximization of the L_1-penalized log-likelihood. For example, Yuan and Lin [13], Banerjee et al.[1] and Dahl et al. [2] adapted interior point optimization methods for the solution to this problem. Both papers revealed that the simpler approach of Meinshausen and Buhlmann [12] can be viewed as an approximation to the exact problem. Banerjee et al. [1] exploited the blockwise coordinate descent approach to solve the lasso problem. Friedman et al. [4] invented the graphical lasso and applied fast coordinate descent algorithms to solve the lasso problem. Graphical lasso is faster than previous methods and also provides a conceptual link between the exact problem and the approximation suggested by Meinshausen et al. [12]. In the following, we introduce the details on how to apply the graphical lasso on QDF.

The graphical lasso estimates the covariance matrix of Gaussian distribution by recursively solving and updating the lasso problem. Suppose, we have N multivariate normal observations of dimension d, with mean μ and covariance Σ. Let $\Theta = \Sigma^{-1}$ and let S be the empirical covariance matrix, the problem of graphical lasso is to maximize the penalized log-likelihood

$$
\log \det \Theta - tr(S\Theta) - \rho \|\Theta\|_1
\tag{6}
$$

Here, tr denotes the trace and $\|\Theta\|_1$ is the L_1 norm$-$the sum of the absolute values of the elements of Σ^{-1}. ρ is a trade-off parameter, which however proves insensitive to the optimization. We set it to 10^{-4} in all the experiments of this paper. Expression (6) is the Gaussian log-likelihood of the data, partially maximized with respect to the mean parameter μ.

Let W be the estimation of Σ. We can solve the problem by optimizing over each row and corresponding column of W in a block coordinate descent approach. Partitioning W and S

$$W = \begin{pmatrix} W_{11} & w_{12} \\ w_{12}^T & w_{22} \end{pmatrix}, S = \begin{pmatrix} S_{11} & s_{12} \\ s_{12}^T & s_{22} \end{pmatrix}, \tag{7}$$

the solution for w_{12} satisfies

$$w_{12} = \arg\min_y \left\{ y^T W_{11}^{-1} y : \|y - s_{12}\|_\infty \le \rho \right\} \tag{8}$$

This is a box-constrained quadratic program (QP), which can be solved using an interior-point procedure. By permuting the rows and columns, the target column is always the last. We can then solve a problem similar to Eq. (8) for each column and update their estimate of W after each stage. This is repeated until convergence. If this procedure is initialized with a positive definite matrix, the iterates from this procedure remains positive definite and invertible, even if $p > N$.

Using convex duality, the solution of problem (8) is equivalent to solving the dual problem

$$\min_\beta \left\{ \frac{1}{2} \left\| W_{11}^{1/2} \beta - b \right\|^2 + \rho \|\beta\|_1 \right\}, \tag{9}$$

where $b = W_{11}^{-1/2} s_{12}$; if β solves Eq. (9), then $w_{12} = W_{11}\beta$ solves Eq. (8). Expression (9) resembles a lasso (L_1 regularized) least squares problem. If $W_{11} = S_{11}$, the solutions $\hat{\beta}$ are easily seen to equal the lasso estimates for the pth variable on the others. When $W_{11} \ne S_{11}$ in general, we can use fast coordinate descent algorithm [3], which makes solution of the lasso problem very attractive.

To solve problem (9), graphical lasso uses W_{11} and s_{12}, where W_{11} is the current estimate of the upper block of W. The algorithm updates w and cycles through all of the variables until convergence.

The detailed algorithm is listed as below:

Algorithm 1. Graphical lasso algorithm

1: Start with $W = S + \rho I$. The diagonal of W remains unchanged in what follows.
2: **for** $j = 1, 2, ...p, 1, 2, ...p, ...$
3:    input: W_{11} and s_{12}
4:    solve the lasso problem (9)
5:    gives a $p - 1$ vector solution $\hat{\beta}$.
6:    fill in the corresponding row and column of W using $w_{12} = W_{11}\hat{\beta}$
7:    continue until convergence
8: **end for**

---

4   Experimental Results

We conduct extensive experiments to verify the effectiveness of the proposed algorithm for covariance estimation and classification. We compare our algorithm to the state-of-the-art algorithm MQDF. All the algorithms are implemented and run using matlab on a PC with 3.0Ghz CPU and 2G RAM.

4.1   Results on Synthetic Data

In this section, we perform experiments on synthetic data to measure how accurate the proposed graphical Lasso covariance estimate will be. We compared the estimated covariance obtained by graphical lasso and the EM algorithm, which is used in QDF. In more details, we first generate samples following a specific Gaussian distribution. We then use EM and Graphical Lasso to estimate the covariance. Finally we examine the estimation error between the ground truth covariance and estimated covariance. The estimation error is computed by Eq. (10)

$$D = sqrt(\sum_{i=1}^{m} \sum_{j=1}^{m} |C_{ij} - C'_{ij}|). \tag{10}$$

We generate both two-dimensional data and ten-dimensional data, the number of samples are from 50 to 10000. The results are listed in Fig. 1.

(a) 2-dimensional estimation   (b) 10-dimensional estimation
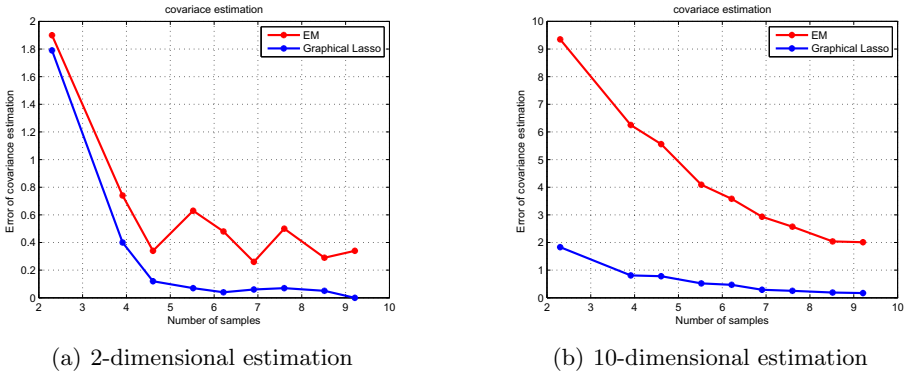
Fig. 1. Estimation Error on Synthetic Data

From the results, we can see that the graphical lasso estimates the covariance more precisely than EM estimator both on 2-dimensional data and 10-dimensional data. The superiority is more distinctive when the number of samples is smaller than the data dimensionality. This can be seen in the left part of Fig. 1(b).

4.2   Results on UCI

To examine the classification performance of GLQDF, we conduct a series of experiments on three data sets from UCI repository: 1)Optdigits: with 10 class and 64 dimension, 3,823 training and 1,797 test samples. 2) Sat, with 6 class and 36 dimension, 4,435 training and 2,000 test samples. 3) HW306class: with 153 class and 512 dimension, 91,365 training and 9,141 test samples. For simplicity, we apply Linear Discriminant Analysis (LDA) to reduce the dimensionality to the class number minus 1 in the experiments. The recognition rate of MQDF

and GLQDF is listed in Table 1. It is clear that the GLQDF achieves better recognition rate in every dataset than MQDF. This clearly demonstrates the advantages of the proposed GLQDF.

Table 1. Recognition rate on UCI data sets

dateset	MQDF	GLQDF
Optdigits	94.0	**94.4**
Sat	84.8	**85.8**
HW306class	93.4	**96.0**

4.3   Results on Handwritten Digital Datasets

In this section, we report the experimental results of the proposed algorithm on two handwritten digital datasets, USPS and MNIST. USPS contains $9,298$ handwriting character measurements divided into 10 classes. The entire USPS data set is divided into two parts, a training set with $7,291$ measurements and a test set with $2,007$ measurements. The original image size is 16×16. The MNIST dataset is another handwritten digits data collection, in which a training set of 60,000 examples and a test set of $10,000$ examples in 10 classes. The original image size is 20×20. We compare the recognition rate of different classifier on both the pixel-level feature and gradient feature. The pixel-level feature number of those two datasets is 256 and 400. The gradient feature is extracted by the algorithm in [9]. We specify 8 direcgions of gradient, choose grid structure of 4×4 for USPS and 5×5 for MNIST. Thus, the gradient feature dimensionality of USPS and MNIST is 128 and 200, respectively. We reduce the dimensionality to c - 1 by LDA in both the USPS and MNIST and feed to the MQDF and GLQDF for training and test. We obtain the hyper-parameter of MQDF, which is a multiplier used for the selection of constant δ_i, by cross validation and we select the principle axes as 8. The final results on pixel feature is listed in Table 2 and the result on gradient feature is listed in Table 3.

From the results, either on the pixel feature or gradient feature, the recognition rate of GLQDF is better than the MQDF. This proves again the effectiveness of the lasso criterion based covariance estimation.

Table 2. Recognition rate on handwritten digits data set of pixel feature

dataset	MQDF	GLQDF
USPS	89.09	**89.74**
MNIST	89.91	**90.07**

Table 3. Recognition rate on handwritten digits data set of gradient feature

dataset	MQDF	GLQDF
USPS	95.96	**96.16**
MNIST	**98.21**	**98.21**

4.4   Results on Handwritten Chinese Character Data

We exploit the CASIA data set for comparison. The CASIA data set, collected by the Institute of Automation, Chinese Academy of Sciences, contains $3,755$ Chinese characters of the level-1 set of the standard GB2312-80, 300 samples per class. We choose 250 samples per class for training and the remaining 50 samples per class for test. To save time, we only selected the first 200 classes from CASIA data for our experiment. Each binary image of CASIA data was firstly normalized to gray-scale image of 64×64 pixels by the bi-moment normalization method [10]. Then the 8-direction gradient direction features were extracted. The resulting 512-dimensional feature vector was projected into a low dimensional subspace learned by the global LDA. All of projected vectors were then fed to the MQDF classifier and GLQDF classifier. The hyper-parameter of MQDF was learned by cross validation and its principle axes was set as 20 in different lower subspace.

To compare the performance between MQDF and GLQDF, we projected the original features into different lower subspace and recorded the recognition rate of the corresponding classifier. The results were listed in the Table 4. From the results, we can see that GLQDF almost achieves the same recognition rate as the MQDF, even when the number of lower subspace is equal to 150.

Table 4. Recognition rate on CASIA data set

Dimensionality	MQDF	GLQDF
LDA = 30	**98.72**	**98.72**
LDA = 50	**99.22**	99.15
LDA = 100	**99.52**	99.46
LDA = 150	99.51	**99.54**

5   Conclusion

In this paper, we engage the graphical lasso method to estimate the covariance and propose a new quadratic method called the Graphical Lasso Quadratic Discriminant Function (GLQDF). By exploiting a coordinate descent procedure for the lasso, GLQDF can estimate the covariance matrix more precisely. We can even compute the inverse of the covariance. This solves the singular problem in covariance estimation, especially when the number of samples is smaller than the dimensionality. Extensive experiments demonstrate that the proposed method can perform better than the competitive methods on two artificial and six real data sets.

References

1. Banerjee, O., El Ghaoui, L., d'Aspremont, A.: Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. The Journal of Machine Learning Research 9, 485–516 (2008)
2. Dahl, J., Roychowdhury, V., Vandenberghe, L.: Maximum likelihood estimation of gaussian graphical models: numerical implementation and topology selection. UCLA preprint (2005)
3. Friedman, J., Hastie, T., Höfling, H., Tibshirani, R.: Pathwise coordinate optimization. The Annals of Applied Statistics 1(2), 302–332 (2007)
4. Friedman, J., Hastie, T., Tibshirani, R.: Sparse inverse covariance estimation with the graphical lasso. Biostatistics 9(3), 432–445 (2008)
5. Friedman, J.: Regularized discriminant analysis. Journal of the American Statistical Association 84(405), 165–175 (1989)
6. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning (2001)
7. Hoffbeck, J., Landgrebe, D.: Covariance matrix estimation and classification with limited training data. IEEE Transactions on Pattern Analysis and Machine Intelligence 18(7), 763–767 (1996)
8. Kimura, F., Takashina, K., Tsuruoka, S., Miyake, Y.: Modified quadratic discriminant functions and the application to chinese character recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (1), 149–153 (1987)
9. Liu, C., Nakashima, K., Sako, H., Fujisawa, H.: Handwritten digit recognition: benchmarking of state-of-the-art techniques. Pattern Recognition 36(10), 2271–2285 (2003)
10. Liu, C., Sako, H., Fujisawa, H.: Handwritten chinese character recognition: alternatives to nonlinear normalization. In: Proc. 7th ICDAR, pp. 524–528 (2003)
11. Liu, C., Sako, H., Fujisawa, H.: Discriminative learning quadratic discriminant function for handwriting recognition. IEEE Transactions on Neural Networks 15(2), 430–444 (2004)
12. Meinshausen, N., Bühlmann, P.: High-dimensional graphs and variable selection with the lasso. The Annals of Statistics 34(3), 1436–1462 (2006)
13. Yuan, M., Lin, Y.: Model selection and estimation in the gaussian graphical model. Biometrika 94(1), 19–35 (2007)

Denial-of-Service Attack Detection Based on Multivariate Correlation Analysis

Zhiyuan Tan[1,2], Aruna Jamdagni[1,2], Xiangjian He[1], Priyadarsi Nanda[1], and Ren Ping Liu[2]

[1] Centre for Innovation in IT Services and Applications (iNEXT),
University of Technology, Sydney, Australia
[2] CSIRO Marsfield, Australia
{Zhiyuan.Tan,Aruna.Jamdagni}@student.uts.edu.au,
{Xiangjian.He,Priyadarsi.Nanda}@uts.edu.au,
ren.liu@csiro.au

Abstract. The reliability and availability of network services are being threatened by the growing number of Denial-of-Service (DoS) attacks. Effective mechanisms for DoS attack detection are demanded. Therefore, we propose a multivariate correlation analysis approach to investigate and extract second-order statistics from the observed network traffic records. These second-order statistics extracted by the proposed analysis approach can provide important correlative information hiding among the features. By making use of this hidden information, the detection accuracy can be significantly enhanced. The effectiveness of the proposed multivariate correlation analysis approach is evaluated on the KDD CUP 99 dataset. The evaluation shows encouraging results with average 99.96% detection rate and 2.08% false positive rate. Comparisons also show that our multivariate correlation analysis based detection approach outperforms some other current researches in detecting DoS attacks.

Keywords: Denial-of-Service Attack, Euclidean Distance Map, Multivariate Correlations, Anomaly Detection.

1 Introduction

Network security has received public concerns with the rapid growth and the prevalence of interconnection among computer systems. It is now under spotlight due to the emergence of more sophisticated attack techniques and easy-access user-friendly attack tools which facilitates any person to easily launch network attacks with little programming knowledge. According to studies [1] and [2], there have been 10,000 new viruses or variant of existing viruses recorded in the year of 2004, and billions of dollars loss has been caused by Denial-of-Service (DoS) attacks over the past few years.

The intention of a DoS attack is to deliberately prevent a victim, such as host, router or entire network, from being accessible or being capable of receiving normal services from the Internet. The availability of network services is seriously threatened

by the continuously increasing number of DoS attacks. Thus effective mechanisms for DoS attack detection are highly demanded.

To maintain the reliability and the availability of network services, research community and industry sector have put a lot of efforts to the development of intrusion detection techniques. As one of the powerful network intrusions, DoS attack has been carefully studied in the intrusion detection research over the last decade. Generally, network intrusion detection can be grouped into two main categories, namely signature-based detection [3] and anomaly-based detection [4]. Benefiting from the principal of detection, which monitors and flags any network activity presenting significant deviation from their normal profiles as a suspicious, anomaly-based detection techniques show more advanced in detecting zero-day intrusions [5].

Therefore, recent works in DoS attack detection mainly focus on anomaly-based techniques, and various detection techniques have been proposed. For example, clustering [6] [7], neural network [8] [9], pattern recognition [10], support vector machine [11], nearest neighbor [12] and statistical detection techniques [13] [14] [15].

However, some of these proposed techniques often suffer high false positive rate since the dependencies and correlations of the features are intrinsically neglected [16]. The other techniques are either invalid to flooding-based DoS attacks [10] or incapable of identifying individual attack packets from a group of samples [15].

To address the aforementioned problems, a Euclidean Distance Map (EDM) based multivariate correlation (second-order statistics) analysis approach is proposed in this paper to discover the relations among features within the observed data objects. Significant changes of these relations indicate occurrences of intrusions. Owing to the computational simplify of Euclidean distance and the extracted valuable correlative information, application of the multivariate correlation analysis makes the DoS attack detection more effective and efficient. It achieves high detection accuracy while retaining a low false positive rate. Moreover, benefiting from the principal of anomaly detection, our DoS attack detection approach is independent on prior knowledge of attack and is capable of detecting both known and unknown DoS attacks.

The rest of this paper is organized as follows. Section 2 provides current work related to our research. Section 3 proposes a novel multivariate correlation analysis approach. Section 4 presents a detailed description on the applications of the proposed multivariate correlation analysis approach in DoS attack detection. Section 5 shows the evaluation results of the proposed approach on KDD CUP 99 dataset and makes some analysis. Finally, conclusions and future work are given in Section 6.

2   Related Work

Owing to the advantage in detecting unknown attacks, anomaly intrusion detection mechanism has captured the major attention from research community. Researchers focused on studies of sequential change-point detection based statistical DoS attack detection approaches [17] [18] in the early 2000s. The approaches make use of the abrupt change occurring in the observed sequential data, such as Management Information Base (MIB) variables and statistics between the number of SYN packets and the number of FIN or SYN/ACK packets. They have been proven effective in detecting any abrupt change in network traffic. However, on one hand, in Thottan and

Ji's approach [17] the operator matrix may need to frequently update its feature set to cover the emergent attacks, and the approach suffers time granularity problem. On the other hand, the approach proposed by Wang et al. [18] only targets on SYN flooding attacks, and its performance is affected by the passive RST packets. Moreover, the two approaches consider only the first-order statistics and ignore the correlative information which is important to detection accuracy.

Recently, the intrusion detection research community started recognizing the importance of the second-order statistics of monitored-network features. Several researches have been conducted to explore the use of the second-order statistics in DoS attack detection. A team of researchers from the Hong Kong Polytechnic University [15] employed the covariance matrices of the sequential samples and proposed a threshold based detection approach to detecting various types of DoS attacks. Travallaee et al. [19] applied Covariance Matrix Sign (CMS) for DoS attack detection. These approaches achieve encouraging detection rates. However, they still suffer from comparative high false positive rates and do not work under the situation where an attack linearly changes all monitored features. In addition, the approaches can only label a group of observed samples as normal or attacks, and cannot identify individual attack packets from the crowd.

Our work also makes use of the idea of change and the second-order statistics. We investigate the change of the correlations between features which are the second-order statistics of the features in a single observed data object. This makes our approach more advanced in detection accuracy and the ability of labeling individual attack packets.

3   Multivariate Correlation Analysis

The behavior of network traffic is reflected by its statistical properties. DoS attack is a type of intrusions attempting to exhaust a victim's resource, and its traffic behaves different from the normal network traffic. Therefore, the statistical properties can be used to reveal the difference. To well present the statistical properties, we propose a novel multivariate correlation analysis approach which employs Euclidean distance for extracting correlative information (named inner correlation) among the features within an observed data object. The detail is given in the following.

Given an arbitrary dataset $X^T = [x_1{}^T \; x_2{}^T \cdots x_n{}^T]$, where $x_i{}^T = [f_1^i \; f_2^i \cdots f_m^i](1 \leq i \leq n)$ represents the ith m-dimensional traffic record. The dataset can be represented in detail as

$$X = \begin{bmatrix} f_1^1 & f_2^1 & \cdots & f_m^1 \\ f_1^2 & f_2^2 & \cdots & f_m^2 \\ \vdots & \vdots & \ddots & \vdots \\ f_1^n & f_2^n & \cdots & f_m^n \end{bmatrix} \tag{1}$$

where f_l^i is the value of the lth feature in the ith traffic record. l and i are varying from 1 to m and from 1 to n respectively.

In order to further explore the inner correlations of the ith traffic record on a multi-dimensional space, the record $x_i{}^T$ is first transformed into a new m-by-m feature matrix x_i' by simply multiplying an m-by-m identity matrix I as shown in (2).

$$x_i^T I = x_i' = \begin{bmatrix} f_1^i & 0 & \cdots & 0 \\ 0 & f_2^i & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & f_m^i \end{bmatrix}_{m \times m} \tag{2}$$

The elements on the diagonal of the matrix x_i' of are the features of the record x_i^T. Each column of the matrix x_i' is a new m-dimensional feature vector denoted by

$$F_j^{iT} = [F_{j,1}^{'i} \; F_{j,2}^{'i} \cdots F_{j,m}^{'i}] \; . \tag{3}$$

where $F_{j,p}^{'i} = 0$ if $j \neq p$ and $F_{j,p}^{'i} = f_j^i$ if $j = p$. The parameters satisfy the conditions of $1 \leq i \leq n, 1 \leq j \leq m$ and $1 \leq p \leq m$. Thus the m-by-m feature matrix x_i' can be rewritten as (4).

$$x_i' = [F_1^i \; F_2^i \cdots F_m^i] \; . \tag{4}$$

Once the transformation is finished, we can apply the Euclidean distance to extract the correlation between the feature vectors j and k in the matrix x_i', which can be defined as

$$ED_{j,k}^i = \sqrt{(F_j^i - F_k^i)^T (F_j^i - F_k^i)}, \tag{5}$$

where $1 \leq i \leq n, 1 \leq j \leq m$ and $1 \leq k \leq m$. However in practice, (5) can be simplified and rewritten as (6) in order to reduce computational complexity.

$$ED_{j,k}^i = \begin{cases} \sqrt{(f_j^i - 0)^2 + (0 - f_k^i)^2}, & j \neq k \\ 0 & , \quad j = k \end{cases} \tag{6}$$

Therefore, the correlations between features in the traffic record x_i^T are defined by a Euclidean Distance Map (EDM) given below.

$$EDM^i = \begin{bmatrix} ED_{1,1}^i & ED_{1,2}^i & \cdots & ED_{1,m}^i \\ ED_{2,1}^i & ED_{2,2}^i & \cdots & ED_{2,m}^i \\ \vdots & \vdots & \ddots & \vdots \\ ED_{m,1}^i & ED_{m,2}^i & \cdots & ED_{m,m}^i \end{bmatrix}. \tag{7}$$

Since the EDM is a symmetric matrix (in which $ED_{j,k}^i = ED_{k,j}^i$) and there is a zero distance from a feature vector to itself $(ED_{j,k}^i = 0, if \; j = k)$, the upper or the lower triangle of the matrix is sufficient to reveal the inner correlations. Hence, the EDM can be simplified and converted into a new inner correlation vector containing only the lower triangle of the EDM as (8).

$$EDM_{lower}^{i}{}^{T} = \left[ED_{2,1}^{i}\ ED_{3,1}^{i}\cdots ED_{m,1}^{i}\ ED_{3,2}^{i}\ ED_{4,2}^{i}\cdots ED_{m,2}^{i}\cdots ED_{m,m-1}^{i}\right]. \tag{8}$$

For the dataset X, its inner correlations can be represented by (9).

$$EDM_{lower}{}^{T} = \left[EDM_{lower}^{1}{}^{T}EDM_{lower}^{2}{}^{T}\cdots EDM_{lower}^{i}{}^{T}\cdots EDM_{lower}^{n}{}^{T}\right]. \tag{9}$$

By making use of the inner correlations, the changes of network behavior caused by DoS attack can be clearly revealed. Additionally, the distance measure facilitates our analysis approach to withstand the issue of linear change of all features.

4   Multivariate Correlation Analysis Based Detection Approach

The objective of this paper is to develop a detection approach that is effective in detecting any known and unknown DoS attacks. Thus, the concept of anomaly-based IDS, which attempts to identify network intrusions by detecting any significant deviations from a profile generated using only normal traffic records in training phase, is the best fit to our problem.

4.1   Norm Profile Generation

In this work, the norm profile is first built through the density estimation of the Mahalanobis Distances (MDs) between observed normal traffic records and the expectation of the normal traffic record. To obtain the distribution of the MDs, two parameters are required to be determined. They are the mean μ and the standard deviation σ of the distances.

Assume that there is a set of g normal training traffic records, which is denoted by $EDM_{lower}^{normal}{}^{T} = \left[EDM_{lower}^{norml,1}{}^{T}\ EDM_{lower}^{normal,2}{}^{T}\cdots EDM_{lower}^{normal,g}{}^{T}\right]$, the parameters can be determined by using the equations shown below. The mean μ is defined as

$$\mu = \frac{1}{g}\sum_{i=1}^{g}MD^{normal,i}, \tag{10}$$

$$MD^{normal,i} = \sqrt{\frac{(EDM_{lower}^{normal,i} - \overline{EDM_{lower}^{normal}})^{T}(EDM_{lower}^{normal,i} - \overline{EDM_{lower}^{normal}})}{Cov}}. \tag{11}$$

The expectation of the lower triangles of normal EDMs ($\overline{EDM_{lower}^{normal}}$) and the covariance matrix (Cov) are given in (12) and (13).

$$\overline{EDM_{lower}^{normal}} = \frac{1}{g}\sum_{i=1}^{g}EDM_{lower}^{normal,i} \tag{12}$$

$$Cov = \begin{bmatrix} \sigma_{ED_{2,1}^{normal}ED_{2,1}^{normal}} & \sigma_{ED_{2,1}^{normal}ED_{3,1}^{normal}} & \cdots & \sigma_{ED_{2,1}^{normal}ED_{m,m-1}^{normal}} \\ \sigma_{ED_{3,1}^{normal}ED_{2,1}^{normal}} & \sigma_{ED_{3,1}^{normal}ED_{3,1}^{normal}} & \cdots & \sigma_{ED_{3,1}^{normal}ED_{m,m-1}^{normal}} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{ED_{m,m-1}^{normal}ED_{2,1}^{normal}} & \sigma_{ED_{m,m-1}^{normal}ED_{3,1}^{normal}} & \cdots & \sigma_{ED_{m,m-1}^{normal}ED_{m,m-1}^{normal}} \end{bmatrix} \tag{13}$$

The covariance between two arbitrary second-order statistics in the lower triangle of normal EDM is as given in (14)

$$\sigma_{ED_{j,k}^{normal}ED_{l,v}^{normal}} = \frac{1}{g-1}\sum_{i=1}^{g}(ED_{j,k}^{normal,i} - \mu_{ED_{j,k}^{normal}})(ED_{l,v}^{normal,i} - \mu_{ED_{l,v}^{normal}}) , \tag{14}$$

where $\mu_{ED_{j,k}^{normal}} = \frac{1}{g}\sum_{i=1}^{g} ED_{j,k}^{normal,i}$. The standard deviation σ can be obtained by using (15).

$$\sigma = \sqrt{\frac{\sum_{i=1}^{g}(MD^{normal,i} - \overline{MD^{normal}})^2}{g-1}} \tag{15}$$

The obtained distribution of the normal training traffic records is stored in the norm profile for attack detection.

4.2   Attack Detection

According to the definition of normal distribution, roughly 99.7% of the values are within 3 standard deviations from the mean. Therefore, the decision can be made by comparing the distance of an observed object to the mean of the distribution. If the distance is greater than 3 standard deviations from the mean, it is flagged as an attack with 99.7% confidence.

To make the comparison, the EDM of the observed traffic record ($EDM_{lower}^{observed}$) needs to be generated using the proposed multivariate correlation analysis approach. Then, the MD between the $EDM_{lower}^{observed}$ and the expectation ($\overline{EDM_{lower}^{normal}}$) of the lower triangles of normal EDMs stored in the normal profile is computed using (16).

$$MD^{observed} = \sqrt{\frac{(EDM_{lower}^{observed} - \overline{EDM_{lower}^{normal}})^T (EDM_{lower}^{observed} - \overline{EDM_{lower}^{normal}})}{Cov}} \tag{16}$$

Then, the values are compared with the pre-defined threshold given in (17).

$$Threshold = \mu + \sigma * n \tag{17}$$

For a normal distribution, n is usually ranged from 1 to 3. This means that we would like to make a detection decision with a certain level of confidence varying from 68% to 99.7% in associate with the selection of different values of n. Therefore, if the observed MD is greater than the threshold, it will be considered as an attack.

5   Evaluation and Analysis

This section describes and analyzes the results obtained from the evaluation of the performance of the proposed multivariate correlation analysis based approach on DoS attack detection. The evaluation is conducted on KDD CUP 99 dataset [20].

The 10 percent labelled dataset of KDD CUP 99 dataset is applied for evaluation purpose. Although the dataset is not without criticism [21], it is the only public dataset with labelled attack samples. Moreover, for the comparison, many research works were evaluated using this dataset. There are six different types of DoS attacks available from the 10 percent labelled dataset. They are Teardrop, Smurf, Pod, Neptune, Land and Back attacks.

To evaluate an IDS, detection rate and false positive rate are two important metrics. We aim to achieve a high detection rate while retaining a low false positive rate. To visually reveal the performance of the IDS, Receiver Operating Characteristic (ROC) curve is employed to show the relations between these two metrics.

5.1   Preprocessing

We first filter all records with the labels of Normal, Teardrop, Smurf, Pod, Neptune, Land and Back from the 10 percent labelled dataset. Then, further classify them into different clusters according to their labels. The description of the filtered data is presented in Table 1.

Table 1. Number of records of normal and DoS attack records

Normal	Teardrop	Smurf	Pod	Neptune	Land	Back
97,260	9,790	2,807,900	2,640	1,072,010	210	22,030

5.2   Results and Analysis

To evaluate the detection performance of the proposed approach, we conduct 10 fold cross-validations and 32 continuous features are used. Norm profiles are built with respect to different types of traffic, namely TCP, UDP and ICMP traffic. In the training phase, we only employ the Normal records, while Normal records and the attack records are all involved in the test phase. In the test phase, we vary the parameter n given in the (17) from 1 to 3 with an increment of 0.5, in order to compare the detection accuracy with the change of the threshold. The results are shown in Table 2.

As can be seen from Table 2, our proposed multivariate correlation analysis based detection approach performs very well in most of the cases. The detection rates of Normal records rise from 97.92% to 99.13% along with the increase of the threshold. The Smurf and Pod attack records are completely detected without being affected by the change of the threshold. For Teardrop and Neptune attacks, our approach achieves approximately 100% almost in all cases. The detection approach suffers from mirror degeneration in the case of Land attack when the threshold set greater than 2σ, but it still manages to detect around 74.76% of the attack records. However, when detecting

Back attacks, the detection accuracy suffers from a sharp decrease from 92.37% to 45.71% and finally down to 6.42% when the threshold keeps increasing. This problem may be caused by the non-normalized data in which some features dominate the detection performance during their comparatively large values.

Table 2. Detection rates for normal and DoS attack records against different thresholds

Type of records	Threshold				
	1σ	1.5σ	2σ	2.5σ	3σ
Normal	97.92%	98.47%	98.75%	98.99%	99.13%
Teardrop	100%	100%	100%	99.99%	99.98%
Smurf	100%	100%	100%	100%	100%
Pod	100%	100%	100%	100%	100%
Neptune	100%	100%	100%	99.99%	99.99%
Land	100%	100%	96.19%	87.62%	74.76%
Back	92.37%	45.71%	9.93%	9.16%	6.42%

To better understand the performance of our proposed multivariate correlation analysis based detection approach, ROC curves are given in Fig. 1.
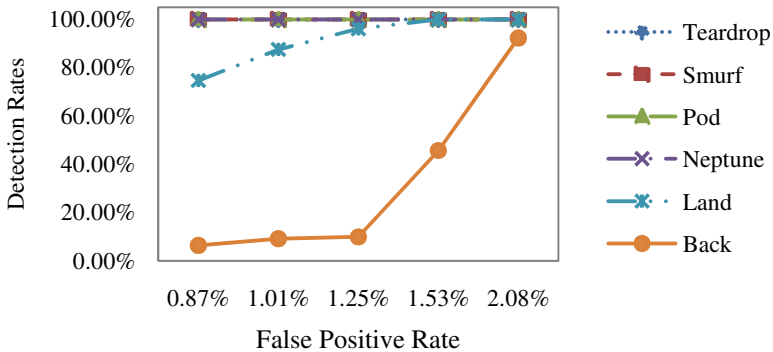
Fig. 1. ROC curves for the detection of DoS attacks

The ROC curves show clear tradeoff between the detection rate and false positive rate. There is a common trend that detection rates of all attacks increase when larger numbers of false positive are tolerated. If high detection rates are required, we have to endure a comparatively high false positive rate of 2.08%. However, 2.08% false positive rate is still in the acceptable range and even better than other detection approaches.

To further evaluate the performance of the proposed approach, a comparison with covariance feature space based network intrusion detection [15] and HCVM [22] is given in Table 3.

Table 3. Performance comparison of different detection approaches

	Multivariate correlation analysis based detection approach (Threshold = 1σ)	Covariance feature space based network intrusion detection (Threshold approach with 3D principle) [15]	HCVM[22]
Detection rate	99.96%	99.95%	93.2%
False positive rate	2.08%	10.33%	5.4%

In general, our multivariate correlation analysis has been proven by the evaluation results and it can correctly extract the statistical properties to exhibit the behaviour of network traffic. The application of multivariate correlation analysis in DoS detection gives promising outcomes.

6   Conclusions and Future Work

This paper has proposed a Euclidean distance based multivariate correlation analysis approach to extract the inner correlations (second-order statistics) of network traffic records, which can better exhibit the network traffic behaviours. We have evaluated the analysis approach using the KDD CUP 99 dataset. The results show that these second-order statistics can clearly reveal the changes of network behavior caused by DoS attack. The multivariate correlation analysis based DoS attack detection approach achieves 99.96% detection rate and 2.08% false positive rate. The detection accuracy is improved by involving the second-order statistics instead of the original first-order statistics into the classification.

However, our approach still suffers from a high false negative rate in detecting Back attack. This may be caused by the non-normalized data or the redundant features in the dataset. Therefore, we will employ data normalization methods and optimal feature selection in our future work in order to improve the detection accuracy. Also, temporal information will be considered in the successive research.

References

1. Kay, J.: Low Volume Viruses: New Tools for Criminals. In: Network Security, pp. 16–18 (2005)
2. Gordon, L.A., Loeb, M.P., Lucyshyn, W., Richardson, R.: 2005 CSI/FBI Computer Crime and Security Survey. Computer Security Journal 21(1) (2005)
3. Roesch, M.: Snort-lightweight Intrusion Detection for Networks. In: The 13th USENIX Conference on System Administration, pp. 229–238. USENIX, Seattle (1999)
4. Patcha, A., Park, J.M.: An Overview of Anomaly Detection Techniques: Existing Solutions and Latest Technological Trends. Computer Networks 51, 3448–3470 (2007)
5. Denning, D.E.: An Intrusion-detection Model. IEEE Transactions on Software Engineering, 222–232 (1987)
6. Jin, C., Wang, H., Shin, K.G.: Hop-count Filtering: An Effective Defense Against Spoofed DDoS Traffic. In: The 10th ACM Conference on Computer and Communications Security, pp. 30–41. ACM (2003)

7. Lee, K., Kim, J., Kwon, K.H., Han, Y., Kim, S.: DDoS Attack Detection Method Using Cluster Analysis. Expert Systems with Applications 34, 1659–1665 (2008)
8. Amini, M., Jalili, R., Shahriari, H.R.: RT-UNNID: A practical solution to real-time network-based intrusion detection using unsupervised neural networks. Computers & Security 25, 459–468 (2006)
9. Wang, G., Hao, J., Ma, J., Huang, L.: A New Approach to Intrusion Detection Using Artificial Neural Networks and Fuzzy Clustering. Expert Systems with Applications 37, 6225–6232 (2010)
10. Jamdagni, A., Tan, Z., Nanda, P., He, X., Liu, R.P.: Intrusion Detection Using GSAD Model for HTTP Traffic on Web Services. In: The 6th International Wireless Communications and Mobile Computing Conference, pp. 1193–1197. ACM (2010)
11. Fugate, M., Gattiker, J.R.: Computer Intrusion Detection with Classification and Anomaly Detection Using SVMs. International Journal of Pattern Recognition and Artificial Intelligence 17, 441–458 (2003)
12. Lane, T., Brodley, C.E.: Temporal Sequence Learning and Data Reduction for Anomaly Detection. ACM Transactions on Information and System Security (TISSEC) 2, 295–331 (1999)
13. Ye, N., Emran, S.M., Chen, Q., Vilbert, S.: Multivariate Statistical Analysis of Audit Trails for Host-based Intrusion Detection. IEEE Transactions on Computers, 810–820 (2002)
14. Manikopoulos, C., Papavassiliou, S.: Network Intrusion and Fault Detection: A Statistical Anomaly Approach. IEEE Communications Magazine 40, 76–82 (2002)
15. Jin, S., Yeung, D.S., Wang, X.: Network Intrusion Detection in Covariance Feature Space. Pattern Recognition 40, 2185–2197 (2007)
16. Sarasamma, S.T., Zhu, Q.A., Huff, J.: Hierarchical Kohonen Net for Anomaly Detection in Network Security. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 35, 302–312 (2005)
17. Thottan, M., Ji, C.: Anomaly Detection in IP Networks. IEEE Transactions on Signal Processing 51, 2191–2204 (2003)
18. Wang, H., Zhang, D., Shin, K.G.: Change-point Monitoring for the Detection of DoS Attacks. IEEE Transactions on Dependable and Secure Computing, 193–208 (2004)
19. Tavallaee, M., Lu, W., Iqbal, S.A., Ghorbani, A.A.: A Novel Covariance Matrix Based Approach for Detecting Network Anomalies. In: The Communication Networks and Services Research Conference, pp. 75–81. IEEE (2008)
20. Cheng, J., Hatzis, C., Hayashi, H., Krogel, M.A., Morishita, S., Page, D., Sese, J.: KDD Cup 2001 report. ACM SIGKDD Explorations Newsletter 3, 47–64 (2002)
21. Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.A.: A Detailed Analysis of the KDD Cup 99 Data Set. In: The Second IEEE International Conference on Computational Intelligence for Security and Defense Applications, pp. 1–6. IEEE (2009)
22. Chen, Y., Pang, S., Kasabov, N., Ban, T., Kadobayashi, Y.: Hierarchical Core Vector Machines for Network Intrusion Detection. In: Leung, C.S., Lee, M., Chan, J.H. (eds.) ICONIP 2009. LNCS, vol. 5864, pp. 520–529. Springer, Heidelberg (2009)

Deep Belief Networks for Financial Prediction

Bernardete Ribeiro[1] and Noel Lopes[1,2]

[1] CISUC, Department of Informatics Engineering, University of Coimbra, Portugal
[2] UDI/IPG - Research Unit, Polytechnic Institute of Guarda, Portugal
`bribeiro@dei.uc.pt, noel@ipg.pt`

Abstract. Financial business prediction has lately raised a great interest due to the recent world crisis events. In spite of the many advanced shallow computational methods that have extensively been proposed, most algorithms have not yet attained a desirable level of applicability. All show a good performance for a given financial setup but fail in general to create better and reliable models. The main focus of this paper is to present a deep learning model with strong ability to generate high level feature representations for accurate financial prediction. The proposed Deep Belief Network (DBN) approach tested in a real dataset of French companies compares favorably to shallow architectures such as Support Vector Machines (SVM) and single Restricted Boltzmann Machine (RBM). We show that the underlying financial model with deep machine technology has a strong potential thus empowering the finance industry.

Keywords: Deep Learning, Neural Networks, Financial Prediction.

1 Introduction

The robustness and efficiency by which humans can recognize objects has since ever been an intriguing challenge in computational intelligence. Motivated by the extreme efficiency of the visual recognition system recent studies in brain science fields show that this is largely due to the expressive deep architecture employed by human visual cortex systems [11]. One important notion is the hierarchical organization of the brain [5] which plays a critical role on understanding how brain works. The Hierarchical Temporal Memory (HTM) is a machine learning model developed by Hawkins [5] that combines ideas to mimic the neocortex where higher hierarchy levels reuse patterns learned at the lower levels by combining them to memorize more complex patterns. HTM combines and extends approaches used in Bayesian networks, spatial and temporal clustering algorithms, while using a layered-shaped hierarchy of nodes that is common in neural networks. With the same goal of finding higher level features to represent concepts other generative models broadly construed on a deep neural architecture were first introduced by Hinton [6,7] who developed the contrastive divergence method for training Deep Belief Networks (DBN). Motivated by the fact that in inference and prediction lies the core activity of all the global brain functions, these generative models have very recently gained popularity giving rise to successful applications [4,11]. An important yet not exploited area is the financial

credit risk. At the heart of the present global recession there is an inappropriate evaluation of credit risk and most of governments were forced to implement rescue plans for the banking system. While in the past the small, medium (and micro) companies had higher propensity of bankruptcy, in the recent past financial distress in large firms (across all industries) is systematically announced. Research has been very intensive in the last years and many approaches have been devised mainly using discriminative techniques [1,9,10]. In particular, the shallow neural network approach has been found useful in predicting corporate distress from financial statements. Given the devastating effects of the financial distress of firms, it is urgent that management and regulators are able to anticipate this kind of issues. Boosting the accuracy of credit risk methodologies used by banks and financial institutions may lead to considerable gains. In this paper, we present instead a deep neural architecture trained by the contrastive divergence method [7] in a financial database containing 780,000 financial statements of French companies.

The remainder of the paper is organized as follows. In Section 2 both the Restricted Boltzmann Machine (RBM) model and the deep learning algorithm for Deep Belief Networks (DBN) are briefly described. The experimental setup is presented in Section 3 which examines the Diane database and gives the evaluation metrics for problem assessment. In Section 4 we analyze the results obtained with the two-step deep learning methodology. Finally in Section 5 we summarize the main conclusions and address lines of future work.

2 Deep Learning Network Models

Theoretical results suggest that deep learning architectures with multiple levels of non-linear operations provide high-level abstractions for object recognition similar to those found in the human brain. Deep Belief Networks (DBN) have recently been proposed by Hinton [7] with notable success excelling the state-of-the-art in visual recognition and AI areas. Bengio [2] gives an overview of the learning algorithms for deep architectures, in particular those exploiting Restricted Boltzmann Machines (RBM), which are used to construct deeper models such as DBN.

An RBM is an energy-based generative model that consists of a layer of binary visible units ($\mathbf{v}$, whose states are observed) and a layer of binary hidden units ($\mathbf{h}$, whose states cannot be observed) [7,8]. The hidden units with no pairwise connections act as latent variables (features) that allow the RBM to model distributions over state vectors (see Figure 1). With these restrictions, the hidden units are conditionally independent given visible units (i.e. a visible vector). Given an energy function $E(\mathbf{v}, \mathbf{h})$ on the whole set of visible and hidden units, the joint probability is given by:

$$p(\mathbf{v}, \mathbf{h}) = \frac{e^{-E(\mathbf{v}, \mathbf{h})}}{Z} \tag{1}$$
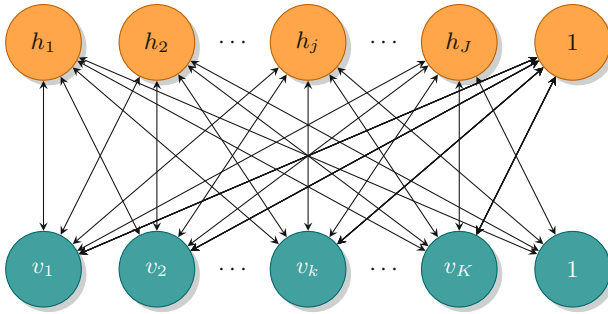
Fig. 1. Restricted Boltzmann Machine (RBM)

where Z is a normalizing partition function i.e., it ensures that $p(\mathbf{v}, \mathbf{h})$ is a valid distribution. For the binary units $h_i \in \{0, 1\}$ and $v_i \in \{0, 1\}$ the energy function of the whole network is:

$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{h}^T \mathbf{W} \mathbf{v} - c^T \mathbf{v} - b^T \mathbf{h}$$
$$= -\sum_{jk} W_{jk} v_k h_j - \sum_k c_k v_k - \sum_j b_j h_j \tag{2}$$

where $\mathbf{W}$ is the matrix of weights, and b and c are the bias units w.r.t. hidden and visible layers, respectively. The marginal distribution over $\mathbf{v}$ is:

$$p(\mathbf{v}) = \sum_h p(\mathbf{v}, \mathbf{h}) = \sum_h p(\mathbf{v}|\mathbf{h}) p(\mathbf{h}) \tag{3}$$

With J hidden units the hidden vector $\mathbf{h}$ can take 2^J possible values, thus 2^J distributions $p(\mathbf{v}|\mathbf{h})$. Therefore, computing the marginal for a large J is impractical. A good estimator of the log-likelihood gradient is the Contrastive Divergence (CD) algorithm [7]. A good property of the RBM is that the posterior of one layer given the other is easy to compute by (4) and (5)

$$p(\mathbf{v}|\mathbf{h}) = \prod_k p(v_k|\mathbf{h}) \text{ where } p(v_k = 1|\mathbf{h}) = sigm(c_k + \sum_j W_{jk} h_j) \tag{4}$$

$$p(\mathbf{h}|\mathbf{v}) = \prod_j p(h_j|\mathbf{v}) \text{ where } p(h_j = 1|\mathbf{v}) = sigm(b_j + \sum_k W_{jk} v_k) \tag{5}$$

where sigm $(\cdot)$ is the sigmoid squashing function $\frac{1}{(1+e^{-z_i})}$ with $z_i = b_i + \sum_j W_{ji} s_j$ where s is the state of the unit j and b is the bias. Inference of hidden factor $\mathbf{h}$ given the observed $\mathbf{v}$ can be easily done because $\mathbf{h}$ is conditionally independent given $\mathbf{v}$.

A Deep Belief Network (DBN) is a generative model with an input layer and an output layer, separated by l layers of hidden stochastic units. The multilayer neural network can efficiently be trained by composing RBMs using the feature
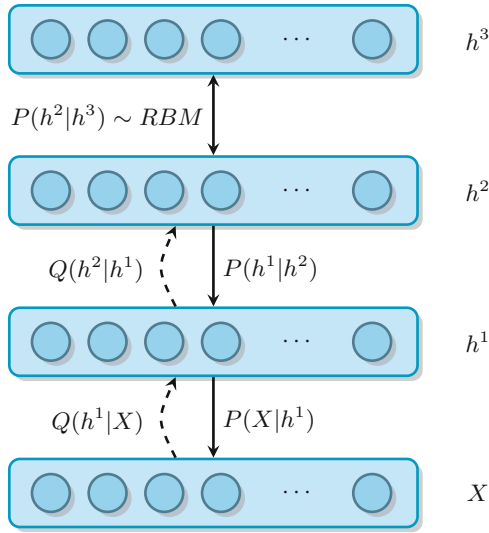
Fig. 2. Deep Belief Network (DBN)

activations of one layer as the training data for the next (see Figure 2). The rationale is that the whole network can be viewed as a single, multilayer generative model and each additional layer improves a lower bound on the probability that the multilayer model would generate the training data [6]. Learning one hidden layer at a time is much more effective given their size which can be very large (MM of weights). Besides, highest level features are much more useful for classification (or dimension reduction) than raw data vectors. An energy-based model of RBMs can be learnt by performing (stochastic) gradient descent on the empirical negative log-likelihood of the training data with respect to the RBM parameters,

$$\frac{\partial}{\partial \theta}(-\log p(\mathbf{v}_0)) = E_{p(\mathbf{h}|\mathbf{v}_0)}\left[\frac{\partial E(\mathbf{v}_0, \mathbf{h})}{\partial \theta}\right] - E_{p(\mathbf{v},\mathbf{h})}\left[\frac{\partial E(\mathbf{v}, \mathbf{h})}{\partial \theta}\right] \tag{6}$$

where θ are the model parameters and E_p is the expectation. This gradient is difficult to compute analytically. Markov Chain Monte Carlo methods are well-suited for RBM models. One iteration of the Markov chain works empirically well and corresponds to the following sampling procedure:

$$\mathbf{v}_0 \xrightarrow{p(\mathbf{h}_0|\mathbf{v}_0)} \mathbf{h}_0 \xrightarrow{p(\mathbf{v}_1|\mathbf{h}_0)} \mathbf{v}_1 \xrightarrow{p(\mathbf{h}_1|\mathbf{v}_1)} \mathbf{h}_1 \tag{7}$$

where the sampling operations are schematically indicated. Estimation of the gradient using the above procedure is denoted by CD-1, where CD-k represents the Contrastive Divergence algorithm [7,3] for performing k iterations of the Markov Chain up to $\mathbf{v}_k$. Given a training set of state vectors (data) learning consists of finding weights and bias that define a Boltzmann distribution in which the training vectors have high probability.

Table 1. Financial Ratios of French DataBase

Variable Description	
x_1 - Number of Employees Previous year	x_{16} - Cashflow / Turnover
x_2 - Capital Employed / Fixed Assets	x_{17} - Working Capital / Turnover days
x_3 - Financial Debt / Capital Employed	x_{18} - Net Current Assets/Turnover days
x_4 - Depreciation of Tangible Assets	x_{19} - Working Capital Needs / Turnover
x_5 - Working Capital / Current Assets	x_{20} - Export
x_6 - Current ratio	x_{21} - Added Value per Employee k EUR
x_7 - Liquidity Ratio	x_{22} - Total Assets Turnover
x_8 - Stock Turnover days	x_{23} - Operating Profit Margin
x_9 - Collection Period days	x_{24} - Net Profit Margin
x_{10} - Credit Period days	x_{25} - Added Value Margin
x_{11} - Turnover per Employee k EUR	x_{26} - Part of Employees
x_{12} - Interest / Turnover	x_{27} - Return on Capital Employed
x_{13} - Debt Period days	x_{28} - Return on Total Assets
x_{14} - Financial Debt / Equity	x_{29} - EBIT Margin
x_{15} - Financial Debt / Cashflow	x_{30} - EBITDA Margin

3   Experimental Setup

We used Diane database which contains financial statements of French compa-
nies. One of the problem goals is the class (healthy, bankrupt) prediction model.
The sample extracted from the initial 780000 statements of the database con-
tained about 60000 financial ratios from industrial French companies (during the
years of 2002 to 2006) with at least 10 employees. From these companies, about
3000 were declared bankrupt in 2007 (or submitted a restructuring plan to the
French court). After pre-processing, the data set contains 1200 French compa-
nies, 600 examples distressed in 2007, and the remainder is healthy. As shown in
Table 1, each instance is characterized by 30 financial ratios produced by Coface[1].
These predictors describe the firms in terms of its financial strength, liquidity,
solvability, productivity of labor and capital, margins, net profitability and re-
turn on investment. In the experiments we took the historical data consisting of
90 inputs spanning three years before bankruptcy. The multilayer unsupervised
learning allows to compactly representing highly non-linear and highly-varying
interrelations among financial status of enterprises thus improving prediction.
The performance metrics defined from the confusion matrix containing the True
Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN)
are as follows: Recall, Precision, Test Accuracy and F1 measure. The latter mea-
sures the trade-off between the Recall and Precision and is a good indicator in
skewed distributions. In financial analysis, the error of type I (False Positive
Rate) indicates a false alarm of the positive class (bankrupt), i.e. it predicts a
healthy company as default. The error of type II (False Negative Rate) indicates
a failure to detect a company with a 'bad' status.

---

[1] Coface is one of largest financial groups in France which provides Credit Insurance,
Factoring Information and Ratings, and Debt Recovery.
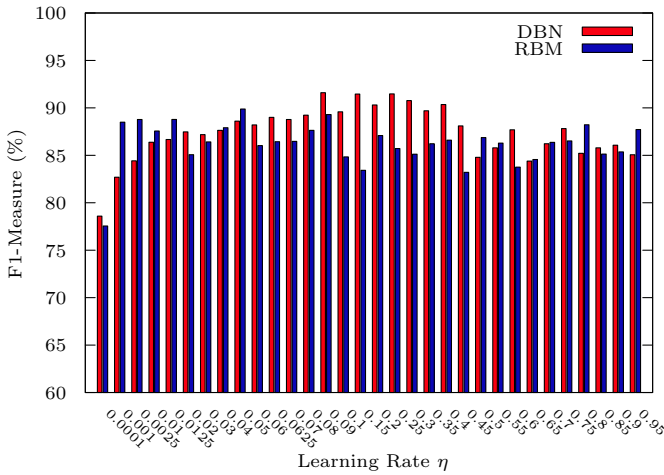
Fig. 3. DBN vs RBM: F1-Measure

4   Results and Discussion

We present the deep learning model for bankruptcy prediction and discuss the results comparing three different methodologies: DBN, RBM and SVM. The data was split in a training and testing data sets with 800 and 400 samples, respectively, for appropriate model selection. The ratios are preprocessed by logarithmized operation to decrease the scatter data distribution. A linear normalization is applied so that the data is transformed to the range between 0 and 1. We put forward a deep architecture model consisting of two steps. In the first step we train the deep multilayer neural network with many levels of non-linearities in a layer-wise unsupervised learning phase. Next, the architecture is refined using a gradient descent based supervised method. The two-stage construction could mostly be used to verify the bankruptcy prediction of a given company. Figure 3 shows the histogram of F1 for both DBN and RBM for the learning rate range [0.0001 − 0.95]. The best mean results were obtained for the learning rate $\eta = 0.094$, momentum $\alpha = 0.3$, penalization constant 2.10^{-3} and max epochs= 500. The configuration of the architecture is composed by 3 hidden layers with nh1= 500, nh2= 500, nh3= 1000 and one output layer with one linear unit. Figure 4 illustrates the mean test error versus the number of epochs for both models. Each experiment was repeated 10 times. The plots for the testing error were obtained by performing a moving average of five points (epochs) to smooth out the learning curves.

Figure 5 illustrates the DBN mean performance w.r.t. F1, test accuracy, type I and type II errors. The best matching F1-score occurs for DBN which shows an improvement by 1.123% and 0.719% over RBM and SVM, respectively. This is also shown in Table 2 which compares the mean and respective standard deviations of the assessment metrics for the tested approaches: DBN, RBM and SVM.
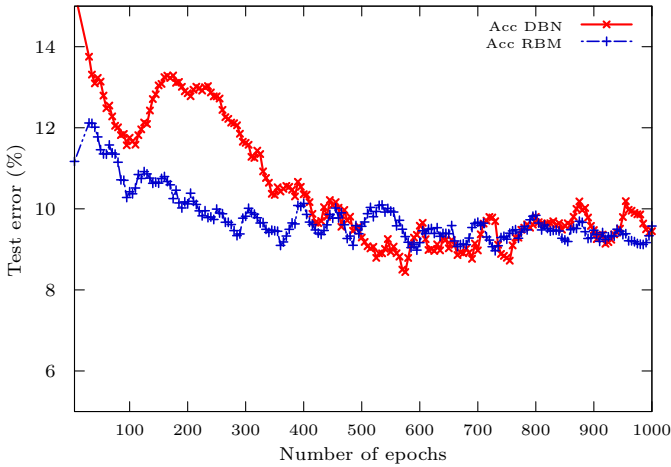
Fig. 4. DBN vs RBM: Test error vs number of epochs
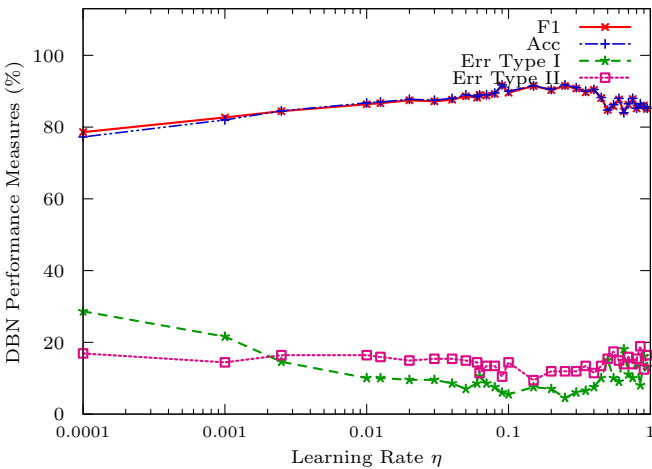
Fig. 5. DBN Performance: F1, Test Accuracy, Type I and Type II errors

The SVM hyperparameters were found by 5-fold cross-validation and RBF kernel was chosen [10]. The Wilcoxon non-parametric test shows that the mean results obtained with DBN are significantly better than RBM (with the same number of parameters). The null hypothesis of the RBM having an equal or better F1 measure than the DBN is rejected at a significance level of 0.05 (with $T = 1.907 > 1.645$) for the range of tested learning rates. It can be gleaned from the results that the deep architecture learns the highly non-linear features at hand given very limited prior knowledge, yielding to an increase of the financial prediction.

Table 2. Testing Performance for Bankruptcy Prediction Models (%)

	Recall	Precision	Type I	Type II	Test Acc	F1
RBM	87.06±1.45	91.09±1.52	8.64±1.72	12.94±1.45	89.20 ± 0.37	89.01±0.36
DBN	87.56±1.69	**93.47±1.97**	**6.23±2.03**	12.44 ±1.69	**90.65± 0.65**	**90.40±0.64**
SVM	**87.67±1.70**	91.95±1.07	7.76±1.10	**12.33±1.70**	89.94±1.04	89.75± 1.11

5   Conclusion and Future Work

We exploited a deep belief network in a financial business problem tested in a database of French companies. Despite the unsupervised characteristic of the model, the last classification stage classifier yields an overall performance that surpasses SVM. We also compared with an RBM in the same database. The preliminary results with a deep learning architecture are promising and raise interest regarding its application to this problem. The properties of the DBN model allow extracting a high-representation of the features that describe the financial status of companies through a greedy layer-wise unsupervised learning. Despite the great prospect of deep learning technologies, to cope with the expensive training of hundred thousands of parameters that need to be adjusted future work will address multicore graphical processors.

References

1. Atiya, A.: Bankruptcy prediction for credit risk using neural networks: A survey and new results. IEEE Trans. on Neural Networks 12(4), 929–935 (2001)
2. Bengio, Y.: Learning deep architectures for AI. Foundations and Trends in Machine Learning 2(1), 1–127 (2009)
3. Bengio, Y., Delalleau, O.: Justifying and generalizing contrastive divergence. Neural Computation 21(6), 1601–1621 (2009)
4. Colbert, R., Weston, J.: A unified architecture for natural language processing: Deep neural networks with multitask learning. In: ICML, pp. 160–167 (2008)
5. Hawkings, J., Blakeslee, S.: On Intelligence. Owl Books, H H & Co., LLC (2004)
6. Hinton, G.E.: Learning multiple layers of representation. Trends in Cognitive Sciences 11, 428–434 (2007)
7. Hinton, G.E., Osindero, S., Teh, Y.: A fast learning algorithm for deep belief nets. Neural Computation 18, 1527–1554 (2006)
8. Larochelle, H., Bengio, Y., Louradour, J., Lamblin, P.: Exploring strategies for training deep neural networks. JMLR 1, 1–40 (2009)
9. Ravi Kumar, P., Ravi, V.: Bankruptcy prediction in banks and firms via statistical and intelligent techniques - a review. European Journal of Operational Research 180(1), 1–28 (2007)
10. Ribeiro, B., Silva, C., Vieira, A., Gaspar-Cunha, A., Neves, J.C.: Financial distress model prediction using SVM+. In: IEEE-IJCNN, pp. 1–7 (2010)
11. Yu, K., Xu, W., Gong, Y.: Deep learning with kernel regularization for visual recognition. In: Neural Information Processing Systems (NIPS), pp. 1889–1896 (2009)

Uncertainty Measure for Selective Sampling Based on Class Probability Output Networks

Ho-Gyeong Kim[1], Rhee Man Kil[2], and Soo-Young Lee[1]

[1] Department of Electrical Engineering
[2] Department of Mathematical Sciences,
Korea Advanced Institute of Science and Technology (KAIST),
291 Daehak-ro, Yuseong-gu, Daejeon 305-701, South Korea
{hogyeong,rmkil,sylee}@kaist.ac.kr

Abstract. This paper presents a novel method of selective sampling using conditional class probabilities estimated from a network referred to as the class probability output network (CPON). For selective sampling, an uncertainty measure is defined using the confidence level for the CPON output. As a result, the proposed uncertainty measure represents how confident the CPON output is. We compared the recognition performance between other sampling methods and the proposed one. The relationship between the uncertainty measure and recognition rate was also investigated.

Keywords: confidence level, uncertainty measure, selective sampling, active learning.

1 Introduction

In many cases of real-world problems of classification or regression, it is expensive to get labels of examples for training. In this situation, the selection of examples is very important in order to obtain a good quality of classifier while reducing the necessary sample size. A possible solution to this problem is to obtain examples selectively which may be helpful to improve the performance of a classifier. We call this method as selective sampling and it is an essential component of active learning that helps that a learner can choose informative training data or queries from the pool. In this selective sampling, the essential component is to define the uncertainty measure; that is, for the given pattern, how uncertain the classifier's decision is. If the given pattern is certain, we don't need to consider that pattern any more since the decision of classification is quite clear. However, if the given pattern is uncertain, we need to collect that pattern for the future training of a classifier.

One interesting approach of defining the uncertainty is to specify a committee [1,2] or an ensemble of hypotheses that are consistent with the training examples [3] and to check the degree of inconsistency with the trained model. However, more natural way of determining uncertain patterns is using the concept of conditional class probability. In this context, we consider to use the class probability output network (CPON) [5] in which the conditional class probability for the given pattern is estimated. In this network, for the uncertainty measure, the concept of confidence intervals for the estimated

statistical parameters is used. For our purpose, the level of confidence for the decision of classification is used for the definition of uncertainty measure. In this way, the uncertainty measure represents how uncertain the classifier's decision is.

The rest of this paper is organized as follows: Section 2 explains the class probability output network estimating the conditional class probabilities for classification problems. Section 3 describes the method of determining the uncertainty measure for selective sampling. Some simulation results to demonstrate the effectiveness of uncertainty measure for selective sampling and learning performance for different sampling methods are investigated in Section 4. Finally, Section 5 concludes this paper.

2   Construction of Class Probability Output Networks

For the construction of CPON, first, we consider the following discriminant function y as the classifier's output for the input pattern $\mathbf{x}$:

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{m} w_i \phi_i(\mathbf{x}|\theta), \tag{1}$$

where m represents the number of kernels and w_i, ϕ_i, and θ represent the ith weight, the ith kernel function, and the kernel parameter, respectively.

We choose the SVM method [7] to obtain the discriminant function since this classifier provides sparse representation of training patterns using the structural risk minimization (SRM) principle. Then, the output of SVM is normalized between 0 and 1 using the linear scaling method. For the normalized classifier's output distributions, the positive and negative classes are approximated by the parameters of the probability distribution. In this training of classifiers, these parameters as well as the kernel parameters are adjusted in such a way that the classifier's output distributions become closer to the ideal distributions.

After the CPON is trained, the classification for an unknown pattern can be determined by the probability distribution for each class. First, for the unknown pattern, the normalized output y of the classifier is computed. Then, the following conditional probability; that is, the output of CPON $F^+(y)$ is calculated:

$$\begin{aligned}
F^+(y) &= P(+|Y^+ \le y \ \ or \ \ Y^- \ge y) \\
&= \frac{F_{Y+}(y)}{F_{Y+}(y) + 1 - F_{Y-}(y)}.
\end{aligned} \tag{2}$$

The final decision can be made using the conditional class probability for the given pattern; that is,

$$\text{class} = \begin{cases} \text{positive if } F^+(y) > 0.5 \\ \text{negative otherwise.} \end{cases} \tag{3}$$

In this way, we can make our decision of pattern classification using the CPON output. For the detail description of applying CPONs to pattern classification problems, refer to [5,8].

3   Uncertainty Measure for Selective Sampling

The suggested CPON provides an effective way of estimating conditional probabilities for classification problems. However, in practice, the conditional class probabilities are estimated from the limited size of data. In this case, the estimated conditional class probabilities might include some uncertainties; that is, the estimated CPON output $\hat{F}^+(y)$ is determined by

$$\hat{F}^+(y) = \frac{\hat{F}_{Y+}(y)}{\hat{F}_{Y+}(y) + 1 - \hat{F}_{Y-}(y)}, \tag{4}$$

where $\hat{F}^+(y)$ and $\hat{F}^-(y)$ represent the estimated CDF values of the positive and negative classes, respectively.

The degree of uncertainty for the decision of classification can be determined by estimating the confidence intervals for the conditional class probabilities. Here, these confidence intervals are estimated from the following K-S statistic [6]:

- First, find the distance measures $D_{n,\alpha}^{\pm}$ for the positive and negative classes; they are

$$D_{n,\alpha}^+ = \frac{K_\alpha}{\sqrt{n^+}} \quad \text{and} \quad D_{n,\alpha}^- = \frac{K_\alpha}{\sqrt{n^-}}, \tag{5}$$

where n^+ and n^- represent the sample size of the positive and negative classes, respectively, and K_α represents the value that satisfies $H(K_\alpha) = 1 - \alpha$, where $H(t)$ is the CDF of the K-S statistic as follows:

$$H(t) = \frac{\sqrt{2\pi}}{t} \sum_{i=1}^{\infty} e^{-(2i-1)^2 \pi^2 / (8t^2)}. \tag{6}$$

- Setting the variables $u^{\pm}$ as follows:

$$u^+ = \hat{F}_{Y+}(y) \quad \text{and} \quad u^- = \hat{F}_{Y-}(y). \tag{7}$$

- Determine the $100(1-\alpha)$ percent confidence intervals for the CDFs of the positive and negative classes:

$$F_{U+}^*(u^+) - D_{n,\alpha}^+ \leq F_{Y+}(y) \leq F_{U+}^*(u^+) + D_{n,\alpha}^+ \tag{8}$$

and

$$1 - F_{U-}^*(u^-) - D_{n,\alpha}^- \leq 1 - F_{Y-}(y) \leq 1 - F_{U-}^*(u^-) + D_{n,\alpha}^-, \tag{9}$$

where $F_{U+}^*(u^+)$ and $F_{U-}^*(u^-)$ represent the empirical CDFs of the uniform distribution for the positive and negative classes, respectively.

The two-sided confidence intervals of (8) and (9) are described by one-sided confidence intervals as follows:

- For the positive class, with a probability of $1 - \alpha/2$,

$$F_{Y+}(y) \leq F_{U+}^*(u^+) + D_{n,\alpha}^+ \quad \text{or} \tag{10}$$
$$F_{Y+}(y) \geq F_{U+}^*(u^+) - D_{n,\alpha}^+. \tag{11}$$

– For the negative class, with a probability of $1 - \alpha/2$,

$$1 - F_{Y^-}(y) \leq 1 - F^*_{U^-}(u^-) + D^-_{n,\alpha} \quad \text{or} \tag{12}$$

$$1 - F_{Y^-}(y) \geq 1 - F^*_{U^-}(u^-) - D^-_{n,\alpha}. \tag{13}$$

Let $F^*_{U^+}(u^+) \geq 1 - F^*_{U^-}(u^-)$. Then, from (11) and (12), we can find the value of α_0 that these two boundaries are met; that is,

$$F^*_{U^+}(u^+) - D^+_{n,\alpha_0} = 1 - F^*_{U^-}(u^-) + D^-_{n,\alpha_0} = x_0. \tag{14}$$

With the above condition,

$$F_{Y^+}(y) \geq x_0 \geq 1 - F_{Y^-}(y). \tag{15}$$

This implies that with a probability of $1 - \alpha_0/2$,

$$F_{Y^+}(y) \geq 1 - F_{Y^-}(y). \tag{16}$$

That is, with a probability of $1 - \alpha_0/2$, the true CDF for the positive class is greater than or equal to the true CDF for the negative class. In the case of $1 - F^*_{U^-}(u^-) \geq F^*_{U^+}(u^+)$, the same argument can be applied using (10) and (13).

From this description of confidence level $1 - \alpha_0/2$, the uncertainty measure δ is determined as follows:

Step 1. From the output of CPON for the positive and negative classes, determine the empirical CDFs $F^*_{U^+}(u^+)$ and $F^*_{U^-}(u^-)$. In the case that the CPON is trained enough for the given patterns,

$$F^*_{U^+}(u^+) \approx \hat{F}_{Y^+}(y) \quad \text{and} \tag{17}$$

$$1 - F^*_{U^-}(u^-) \approx 1 - \hat{F}_{Y^-}(y). \tag{18}$$

Step 2. From the boundary condition x_0, determine the value of K_{α_0}:
  – If $F^*_{U^+}(u^+) \geq 1 - F^*_{U^-}(u^-)$,

$$K_{\alpha_0} = \frac{F^*_{U^+}(u^+) + F^*_{U^-}(u^-) - 1}{1/\sqrt{n^+} + 1/\sqrt{n^-}}. \tag{19}$$

  – Otherwise,

$$K_{\alpha_0} = \frac{1 - F^*_{U^+}(u^+) - F^*_{U^-}(u^-)}{1/\sqrt{n^+} + 1/\sqrt{n^-}}. \tag{20}$$

Step 3. From the value of K_{α_0}, determine the uncertainty value δ_0 as $\alpha_0/2$, where

$$\alpha_0 = 1 - H(K_{\alpha_0}). \tag{21}$$

This uncertainty measure δ represents how well the CDF values of the positive and negative classes are separated. This value usually lies between 0 and 0.5. For example, if the uncertainty value δ_0 is 0.05, it represents that with a probability of $1 - 0.05$ (=0.95), one CDF value is greater than or equal to another CDF value. In this case,
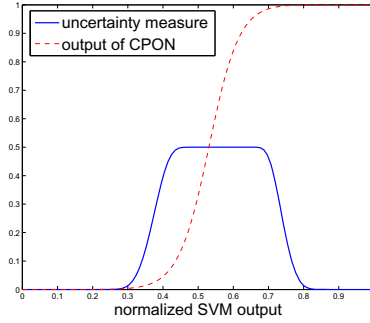
Fig. 1. The uncertainty measure and CPON output plots for the normalized SVM output in which the SVM is trained for the Breast Cancer data set of the UCI database

we presume that the decision of classification is quite certain. On the other hand, if the value is near 0.5, the decision of classification is not quite clear and the given pattern can be treated as an uncertain pattern.

To illustrate the aforementioned uncertainty measure, the uncertainty measure and CPON output plots for the normalized SVM output are shown in Figure 1. Theses plots are obtained from the CPON for binary classification which is trained for the Wisconsin Breast Cancer data set of the University of California at Irvine (UCI) database [9]. In this training, the whole data are used to construct a CPON. As expected, the values of uncertainty measure are larger near the decision boundary and smaller near 0 or 1. This implies that we need to select uncertain examples that are located near the decision boundary.

4 Simulation Results

All the simulations were done with a "LIBSVM" [10], one of SVM libraries for construction of CPONs. For training SVMs, a RBF kernel was used for each construction of a CPON.

4.1 The Effectiveness of Uncertainty Measure

To show the effectiveness of uncertainty measure to be used in the selective sampling, the recognition rate was investigated according to the each value of uncertainty measure. For the benchmark data, we used the UCI database [9]. A brief description of the

Table 1. Description of the data sets from the UCI database

Data Name	Size of Data	Input Dimension	Number of Classes
Diabetes	768	8	2
Breast Cancer	683	10	2
Liver Disorders	345	6	2
Ionosphere	351	34	2
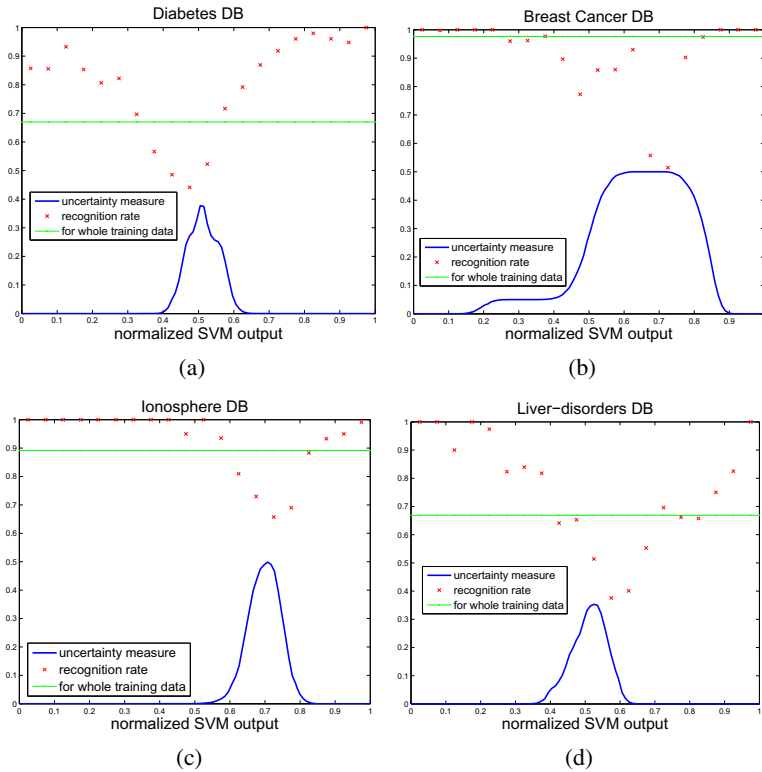
Fig. 2. The uncertainty measure and recognition rates plot for the normalized SVM output in which the SVM is trained for several data set of the UCI database

data sets is given in Table 1. Each attribute of data set was linearly scaled between -1 and 1. For each data set, 50% of data were used as training data and the rest 50% of data were used as test data. With this ratio of training, 10 sets of data were randomly selected and used for training of CPONs. Then, the average value of uncertainty measures and recognition rates was obtained using the trained CPONs and the test data sets, respectively. These simulation results are illustrated in Figures 2(a) through 2(d). In these results, the recognition rate for each output of a classifier showed some variations since the estimation of the recognition rate was different depending on the number of data included in the bin of classifier's output. However, even in these conditions, the results show that 1) the uncertainty measure has a bell-shaped and non-symmetrical figure with different width depending on the data set, especially 2) in the region of larger values of uncertainty measure.

4.2   Comparison of Recognition Performance between Different Sampling Methods

For comparison of recognition performanc, we compared three different sampling methods, random sampling, margin-based sampling [4], and combined sampling with
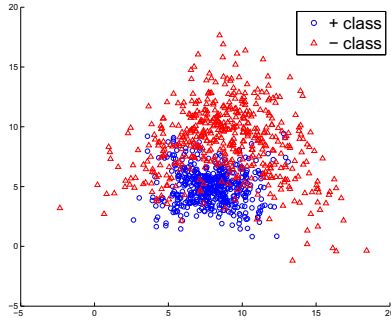
Fig. 3. Synthetic two-dimensional binary data set. Blue circular patterns indicate positive class, and red triangular patterns indicate negative class. Each class has 500 patterns.
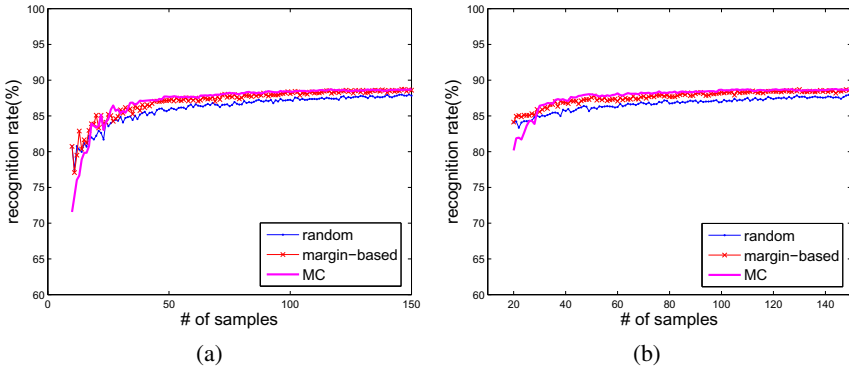
Fig. 4. Comparison of recognition performance along learning epochs, (a) n = 10 (b) n = 20

margin-based and uncertainty measure using confidence level (MC) sampling. Margin-based sampling selects a pattern whose margin is the smallest from the decision boundary. Combined sampling means that initially we used margin-based sampling, and after some epochs used our proposed uncertainty measure using confidence level.

We generated two-dimensional synthetic data set in Figure 3 that has binary balanced classes. Each class has 500 patterns and we initially divided whole data into training and test data with 70% and 30%, respectively. From training data, n data were randomly selected initially; n was set to 10 or 20. The remaining patterns among the training data were regarded as the unlabeled data U. For each learning epoch, we augmented 1 to the current labeled data after labeling, which is selected from the unlabeled data U, until the whole number of the augmented patterns is equal to 150. For MC sampling, the replacing learning epochs is set to 60. After updating labeled and unlabeled data, a classifier was re-trained, and we investigated the recognition rate with test data, and averaged it over 50 trials. As shown in Figure 4, the recognition rate of MC sampling ranked best along learning epochs, compared to the random sampling and margin-based sampling even though initially using margin-based sampling only

for some learning epochs. In the case of ($n = 10$ as illustrated in Figure 4(a), it has fluctuation in the initial learning part due to the small sample size of the training data.

5   Conclusion

We have presented a new way of determining the values of uncertainty measure using the confidence level for the CPON output and a method of selective sampling based on the proposed uncertainty measure. The proposed uncertainty measure represents how uncertain the classifiers decision is. Through the simulation for classification problems using the CPON, we have shown that the proposed uncertainty measure is effective to determine the uncertain region of the classifiers output by comparing the value of uncertainty measure. For selective sampling, the patterns which are mapped into the uncertain region of the classifiers output can be classified as uncertain patterns and these patterns will be selected for the future training of a classifier. Also, we compared the recognition performance compared to other sampling methods. In this way, the proposed uncertainty measure can be easily applied to active learning in which the learning algorithm has the control over the selection of examples for the future training.

References

1. Seung, H., Opper, M., Sompolinski, H.: Query by committee. In: Proceedings of the ACM Workshop on Computational Learning Theory, vol. 452, pp. 287–294 (1992)
2. Freund, Y., Seung, H., Shamir, E., Tishby, N.: Selective sampling using the query by committee algorithm. Machine Learning 28, 133–168 (1997)
3. Krogh, A., Vedelsby, J.: Neural network ensembles, cross validation, and active learning. In: Proceedings of the Neural Information Processing, vol. 7, pp. 231–238 (1994)
4. Tong, S., Koller, D.: Support Vector Machine Active Learning with Applications to Text Classification. In: Proceedings of the Seventeenth International Conference on Machine Learning, pp. 999–1006 (2000)
5. Park, W.: Rhee, K.: Pattern classification with class probability output network. IEEE Transactions on Neural Networks 20(10), 1659–1673 (2009)
6. Rohatgi, V., Saleh, A.: Nonparametric statistical inference. In: An Introduction to Probability and Statistics, 2nd edn. Wiley-Interscience, New York (2001)
7. Vapnik, V.: Statistical Learning Theory. Wiley, New York (1998)
8. Harvey, R., Rhee, K., Han, S.: Automatic media data rating based on class probability output networks. IEEE Transactions on Consumer Electronics 56(4), 2296–2302 (2010)
9. Murphy, P., Aha, D.: UCI Repository of Machine Learning Databases. University of California, Irvine, http://www.archive.ics.uci.edu/ml
10. Chih-Chung, C., Chih-Jen, L.: LIBSVM: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, http://www.csie.ntu.edu.tw/~cjlin/libsvm

Author Index