

Michael Cree
Fay Huang
Junsong Yuan
Wei Qi Yan (Eds.)

Communications in Computer and Information Science

1180

Pattern Recognition

ACPR 2019 Workshops
Auckland, New Zealand, November 26, 2019
Proceedings

 Springer

EXTRAS ONLINE

Communications in Computer and Information Science

1180

Commenced Publication in 2007

Founding and Former Series Editors:

Phoebe Chen, Alfredo Cuzzocrea, Xiaoyong Du, Orhun Kara, Ting Liu,
Krishna M. Sivalingam, Dominik Ślęzak, Takashi Washio, Xiaokang Yang,
and Junsong Yuan

Editorial Board Members

Simone Diniz Junqueira Barbosa 

*Pontifical Catholic University of Rio de Janeiro (PUC-Rio),
Rio de Janeiro, Brazil*

Joaquim Filipe 

Polytechnic Institute of Setúbal, Setúbal, Portugal

Ashish Ghosh

Indian Statistical Institute, Kolkata, India

Igor Kotenko 

*St. Petersburg Institute for Informatics and Automation of the Russian
Academy of Sciences, St. Petersburg, Russia*

Lizhu Zhou

Tsinghua University, Beijing, China

More information about this series at <http://www.springer.com/series/7899>

Michael Cree · Fay Huang ·
Junsong Yuan · Wei Qi Yan (Eds.)

Pattern Recognition

ACPR 2019 Workshops
Auckland, New Zealand, November 26, 2019
Proceedings

Editors

Michael Cree
University of Waikato
Hamilton, New Zealand

Junsong Yuan
State University of New York at Buffalo
Buffalo, NY, USA

Fay Huang
National Ilan University
Yilan, Taiwan

Wei Qi Yan
Auckland University of Technology
Auckland, New Zealand

ISSN 1865-0929 ISSN 1865-0937 (electronic)
Communications in Computer and Information Science
ISBN 978-981-15-3650-2 ISBN 978-981-15-3651-9 (eBook)
<https://doi.org/10.1007/978-981-15-3651-9>

© Springer Nature Singapore Pte Ltd. 2020

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd. The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Preface

The 5th Asian Conference on Pattern Recognition (ACPR 2019), held in Auckland, New Zealand, during November 26–29, 2019, was accompanied by a series of five high-quality workshops covering the full range of state-of-the-art research topics in pattern recognition and related fields.

The workshops consisted of one full-day workshop and four half-day workshops and took place on November 26. Their topics diversely ranged from well-established areas to novel current trends: computer vision for modern vehicles; advances and applications on generative deep learning models (AAGM); image and pattern analysis for multidisciplinary computational anatomy; multi-sensor for action and gesture recognition (MAGR); and towards an automatic data processing chain for airborne and spaceborne sensors.

All submitted papers underwent a double-blind peer-review process, where each paper was reviewed by at least three area experts. Eventually, 23 oral presentations were selected by the individual Workshop Committee with the average acceptance rate of 50%. Additionally, 12 invited talks hosted by individual workshops greatly contributed to the success of the ACPR 2019 workshops. We thank everyone involved in the remarkable programs, committees, reviewers, and authors, for their distinguished contributions.

We hope that you will enjoy reading these contributions, which may inspire your research.

November 2019

Michael Cree
Fay Huang
Junsong Yuan

Organization

Steering Committee

Seong-Whan Lee	Korea University, South Korea
Cheng-Lin Liu	Chinese Academy of Sciences, China
Sankar K. Pal	Indian Statistical Institute, India
Tieniu Tan	Chinese Academy of Sciences, China
Yasushi Yagi	Osaka University, Japan

General Chairs

Reinhard Klette	Auckland University of Technology, New Zealand
Brendan McCane	University of Otago, New Zealand
Umapada Pal	Indian Statistical Institute, India

Program Chairs

Gabriella Sanniti di Baja	Institute of High Performance Computing and Networking, Italy
Shivakumara Palaiahnakote	University of Malaya, Malaysia
Liang Wang	Chinese Academy of Sciences, China

Publication Chair

WeiQi Yan	Auckland University of Technology, New Zealand
-----------	--

International Liaison Chairs

Chokri Ben Amar	University of Sfax, Tunisia - Africa
Wang Han	Nanyang Technology University, Singapore - Asia
Edwin Hancock	University of York, UK - Europe
Anil K. Jain	University of Michigan, USA - North America
Domingo Mery	Pontificia Univerisdad Catolica, Chile - Latin America

Workshop Chairs

Michael Cree	University of Waikato, New Zealand
Fay Huang	National Ilan University, Taiwan
Junsong Yuan	State University of New York at Buffalo, USA

Tutorial Chairs

Michael Blumenstein	University of Technology Sydney, Australia
Yukiko Kenmochi	French National Centre for Scientific Research, France
Ujjwal Maulik	Jadavpur University, India

Sponsorship Chair

Koichi Kise	Osaka University, Japan
-------------	-------------------------

Local Organizing Chair

Martin Stommel	Auckland University of Technology, New Zealand
----------------	--

Organizing Committee

Terry Brydon	Auckland University of Technology, New Zealand
Tapabrata Chakraborty	University of Otago, New Zealand
Gisela Klette	Auckland University of Technology, New Zealand
Minh Nguyen	Auckland University of Technology, New Zealand

Web Manager

Andrew Chen	The University of Auckland, New Zealand
-------------	---

Chairs for the Workshop on Computer Vision for Modern Vehicles

Jinsheng Xiao	Wuhan University, China
Atsushi Imiya	Chiba University, Japan
Hsiang-Jen Chien	Auckland Transport, New Zealand
Thomas Braeunl	The University of Western Australia, Australia

Chairs for the Workshop Advances and Applications on Generative Deep Learning Models

Mariano Rivera	Center for Research in Mathematics AC, Mexico
Wei Qi Yan	Auckland University of Technology, New Zealand
Wangmeng Zuo	Harbin Institute of Technology, China
Adrián Pastor Lopez-Monroy	Center for Research in Mathematics AC, Mexico

Chairs for the Workshop Image and Pattern Analysis for Multidisciplinary Computational Anatomy

Atsushi Imiya

Harvey Ho

Yukiko Kenmochi

Kensaku Mori

Hidekata Hontani

Chiba University, Japan

The University of Auckland, New Zealand

Université Pairs-Est Marne-La-Vallée, France

Nagoya University, Japan

Japan

Chairs for the Workshop Multi-Sensor for Action and Gesture Recognition

Jianyu Yang

Zhigang Tu

JingJing Meng

Soochow University, China

Wuhan University, China

State University of New York, USA

Chairs for the Workshop Towards an Automatic Data Processing Chain for Airborne and Spaceborne Sensors

Ralf Reulke

Bin Luo

Humboldt-Universität zu Berlin, Germany

Wuhan University, China

Contents

Computer Vision for Modern Vehicles

Lane Detection Based on Histogram of Oriented Vanishing Points	3
<i>Shizeng Chen, Bijun Li, Yuan Guo, and Jian Zhou</i>	
Bypass Enhancement RGB Stream Model for Pedestrian Action Recognition of Autonomous Vehicles	12
<i>Dong Cao and Lisha Xu</i>	
Paved and Unpaved Road Segmentation Using Deep Neural Network	20
<i>Dabeen Lee, Seunghyun Kim, Hongjun Lee, Chung Choo Chung, and Whoi-Yul Kim</i>	
Image Scene Conversion Algorithm Based on Generative Adversarial Networks	29
<i>Honggang Xie, Jinsheng Xiao, Junfeng Lei, Wenjuan Xie, and Reinhard Klette</i>	
An Embedded Real-Time Monocular SLAM System Utilizing a Dynamically Reconfigurable Processor	37
<i>Koki Kawashima and Koyo Katsura</i>	
Writer Identification Based on Combination of Bag of Words Model and Multiple Classifiers	47
<i>Ayixiamu Litifu, Yuchen Yan, Jinsheng Xiao, Hao Jiang, Weiqing Yao, and Jihua Wang</i>	
Advances and Applications on Generative Deep Learning Models	
Vehicle-Related Scene Understanding Using Deep Learning	61
<i>Xiaoxu Liu, Minh Neuyen, and Wei Qi Yan</i>	
Spatiotemporal Saliency Based Multi-stream Networks for Action Recognition	74
<i>Zhenbing Liu, Zeya Li, Ming Zong, Wanting Ji, Ruili Wang, and Yan Tian</i>	
Attention Guided Unsupervised Image-to-Image Translation with Progressively Growing Strategy	85
<i>Yuchen Wu, Runtong Zhang, and Keiji Yanai</i>	

Human Motion Generation Based on GAN Toward Unsupervised
3D Human Pose Estimation 100
Sasuke Yamane, Hirotake Yamazoe, and Joo-Ho Lee

vi-MoCoGAN: A Variant of MoCoGAN for Video Generation of Human
Hand Gestures Under Different Viewpoints 110
Thanh-Hai Tran, Viet-Dung Bach, and Huong-Giang Doan

Image and Pattern Analysis for Multidisciplinary Computational Anatomy

Weakly Supervised Domain Adaptation with Point Supervision
in Histopathological Image Segmentation 127
Shun Obikane and Yoshimitsu Aoki

Blood Vessel Enhancement in Liver Region from a Sequence
of Angiograms Taken under Free Breathing 141
*Morio Kawabe, Yuri Kokura, Takashi Ohnishi, Kazuya Nakano,
Hideyuki Kato, Yoshihiko Ooka, Tomoya Sakai, and Hideaki Haneishi*

Real-Time Morphing of the Visible Man Liver
with Intrahepatic Vasculatures 150
Maxime Berg, Changwei Zhang, and Harvey Ho

Development of 3D Physiological Simulation and Education Software
for Pregnant Women 160
Aurélien Bourgeois, Sarah Ancé, and Harvey Ho

Resolution Conversion of Volumetric Array Data for Multimodal
Medical Image Analysis. 169
Kento Hosoya, Kouki Nozawa, and Atsushi Imiya

Multi-sensor for Action and Gesture Recognition

Learning Spatiotemporal Representation Based on 3D Autoencoder
for Anomaly Detection 187
Yunpeng Chang, Zhigang Tu, Bin Luo, and Qianqing Qin

Multi-view Discriminant Analysis for Dynamic Hand Gesture Recognition. . . 196
*Huong-Giang Doan, Thanh-Hai Tran, Hai Vu, Thi-Lan Le,
Van-Toi Nguyen, Sang Viet Dinh, Thi-Oanh Nguyen, Thi-Thuy Nguyen,
and Duy-Cuong Nguyen*

Human Action Recognition Based on Dual Correlation Network. 211
*Fei Han, Dejun Zhang, Yiqi Wu, Zirui Qiu, Longyong Wu,
and Weilun Huang*

Feature Engineering Workflow for Activity Recognition
 from Synchronized Inertial Measurement Units 223
*A. W. Kempa-Liehr, Jonty Oram, Andrew Wong, Mark Finch,
 and Thor Besier*

**Towards an Automatic Data Processing Chain for Airborne
 and Spaceborne Sensors**

Infrared-Image Processing for the DLR FireBIRD Mission 235
*Winfried Halle, Christian Fischer, Thomas Terzibaschian, Adina Zell,
 and Ralf Reulke*

Temperature Dependence of Dark Signal for Sentinel-4 Detector. 253
Ralf Reulke, Michael P. Skegg, and Rüdiger Hohn

An Extended Stochastic Cloning Method for Fusing
 Multi-relative Measurements. 263
Hongmou Zhang, Dirk Baumbach, Denis Grießbach, and Anko Börner

Author Index 277

Computer Vision for Modern Vehicles



Lane Detection Based on Histogram of Oriented Vanishing Points

Shizeng Chen, Bijun Li^(✉), Yuan Guo, and Jian Zhou

State Key Laboratory of Information Engineering in Surveying, Mapping,
and Remote Sensing, Wuhan University, Wuhan 430079, China
{csz, lee, GuoYuan, JianZhou}@whu.edu.cn

Abstract. As an important role in autonomous vehicles or advanced driving assistance systems, lane detection uses the onboard camera high up on the windshield to provide the vehicle's lateral offset within its own lane in a real-time, low-cost way. In this paper, we propose an efficient, robust lane detection method based on histogram of oriented vanishing points. First, the lane features are extracted by symmetrical local threshold. Then, the lines are generated from oriented vanishing points. The lines crossing most features are selected and oriented vanishing points are updated by the overlap between features and selected lines. Last step will be repeated for getting stable oriented vanishing points. Therefore, the last selected lines are most likely to be lane lines. Finally, Validate and select the best lane lines. The proposed method has been tested on a public dataset. The experimental results show that the method can improve robustness under real-time automated driving.

Keywords: Lane detection · Histogram · Oriented vanishing points

1 Introduction

Automated driving is considered to be effective in avoiding driving accidents and improving traffic safety. The lane detection provides basic structural information, guidance information of lane and the relative lateral offset of vehicle to lane [1], which is an indispensable part of the automated driving.

Up to now, lane detection has been widely used in the advanced driving assistance systems (ADAS) to implement lane departure warning systems (LDWS). These products can obtain stable and reliable lane results from standard structured roads such as highways with clear markings and good illumination conditions. However, in order to apply this technology to autonomous vehicles, it must be ensured that it is able to obtain stable and reliable results from roads with complicated illumination conditions, shadows, stains, and various road shapes. Meanwhile, real-time processing is a must.

The typical lane detection algorithm can be divided into three steps: feature extraction; estimating the geometric model of the lane; and tracking the model parameters [2].

The reliability and robustness of feature extraction results directly affect the performance of model estimation and tracking. The commonly used features are edge features and line features. Most of edge point feature extraction is based on underlying

visual features such as color, gradient, grayscale and so on. Steerable filter [3], adaptive threshold [4] or local threshold [5] in scanlines are all based on gradient and grayscale to extract edge points. These methods utilize the fact that the lane markings have significant gradients between the roads. Most of these methods can extract the edge features in a linear time, but the edge features lost a large number of points on lane markings, the information density is low, and the extraction results are susceptible to noise. In the case of shadows or stains, the edge features may require a lot of works to analyze the real edges of the lane lines which increase complexity. For suppressing noises, the performance of the existing denoising methods depends heavily on the accuracy of estimated noise parameters [6]. However, the various road environment means various noises, the feature extraction or the lane detection method should be robust enough.

Lane detection algorithms based on bird's-eye view are also popular [3, 7–12]. In the bird's-eye view, compared with the perspective of monocular camera, the parallel relationship of lane lines, lane position and lane width have obvious consistency, which provides convenience for postprocessing [13]. However, with the pitch, roll or yaw changed caused by road slope or vehicle motion, there may have some distortions of lanes under bird's-eye view. And the process of generating the bird's-eye view is in a high computational complexity.

Aim at the above problems, we propose a real-time and robust lane detection method based on histogram of oriented vanishing points, which uses the orientations of vanishing points instead of the bird's-eye view.

This paper is organized as follows. Section 2 introduce the proposed lane detection method. Section 3 shows the experimental results of our method and comparisons with some other researches. Conclusion is in Sect. 4.

2 Our Method

The overall framework of our method is shown in Fig. 1. Firstly, the image is binarized with the extracted features, and the feature points on the lane are preserved as much as possible. Then, the image is divided into left and right parts according to the center of

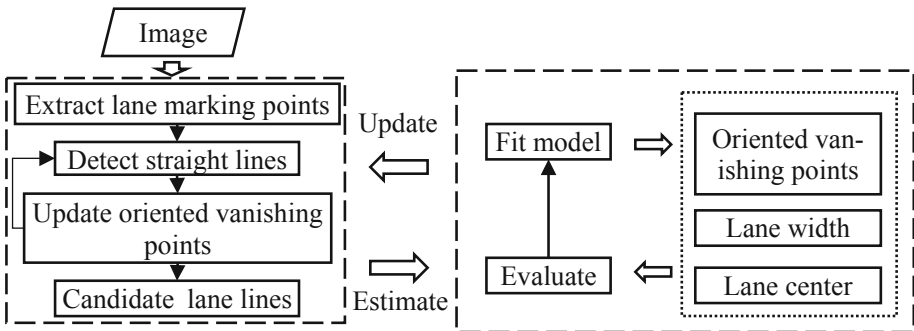


Fig. 1. Overview of our method

the lane and the oriented vanishing points. The oriented vanishing points of left and right lines are respectively tracked. A statistical histogram is constructed based on the oriented vanishing points to extract straight lines. Based on the rough extracted lines which nearly cover the actual lane lines, the oriented vanishing points of the current frame are searched and updated by the overlap between lines and features. Then, re-detected the straight lines and re-updated oriented vanishing points until the oriented vanishing points are stable. So that, the candidate lane lines are extracted based on the detected lines. Finally, the lane lines are estimated and modeled as straight lines with the lane width, the lane center and the oriented vanishing points.

2.1 Feature Extraction

In our method, the Symmetrical Local Threshold (SLT) [2] with lane width constraint is introduced to extract the feature points of the lane markings.

$$\left\{ \begin{array}{l} S_{left}(x) = S_{left}(x-1) + I_{scan}(x) - I_{scan}(x-s) \\ S_{right}(x) = S_{right}(x-1) - I_{scan}(x) + I_{scan}(x+s) \\ Grad_{left}(x) = I_{scan}(x) - \frac{S_{left}(x)}{s} \\ Grad_{right}(x) = I_{scan}(x) - \frac{S_{right}(x)}{s} \end{array} \right. \quad (1)$$



Fig. 2. (a) The grayscale image. (b) The result of feature image in binary.

The grayscale image is scanned line by line. Using $I_{scan}(x)$ to indicate the image row to be scanned now, $Grad_{left}$ and $Grad_{right}$ are got by Eq. 1 which show the gradients between the grayscale in index x and the average grayscales in $[x-s, x)$, $(x, x+s]$. Commonly, the s is a multiple of the lane line width which is calculated in proportion according to the perspective effect. Then, mark the binary values of pixel in where $Grad_{left} > T$ and $Grad_{right} > T$ as 1, otherwise as 0 and the binary image $I_{bin}(row, col)$ is constructed. The results are shown in Fig. 2.

To ensure the integrity of the lane information analyzed by the next histogram step, more lane markings should be extracted. So that, the T selected is smaller than commonly used.

2.2 Line Detection and Oriented Vanishing Points Update

It is a common practice in the bird's-eye view [3, 7–10] to scan column by column to locate the initial position of the lane lines or generate a probability map. Similarly, the proposed method also scans image column by column, but in a camera perspective. This step is completed by vanishing points.

The lane detection methods [15, 16] base on vanishing points usually assume that the lane lines are parallel to each other, then extracting lines and voting in some way to get the vanishing points. Although the lane lines are parallel to each other in many environments, the lane lines will be in non-parallel state for some conditions such as up and down ramps and lane narrowing in road intersection. Such vanishing estimation methods are no applicable. However, the two boundaries of the lane line are parallel. So, the propose method considers left and right lane lines respectively.

As Fig. 3(a) shows, there are two points defined as oriented vanishing points (x_{vl}, y_v) , (x_{vr}, y_v) where horizon line y_v is a constant value initialed by camera model [14] to indicate the orientations of left and right lane lines. Then, lines are generated from x_{vl} and x_{vr} to image bottom y_b in range of $|x - x_{lane}| < w_{lane}$, where w_{lane} is the width of ego-lane and x_{lane} is the center of ego-lane which will be updated by detected lane lines.

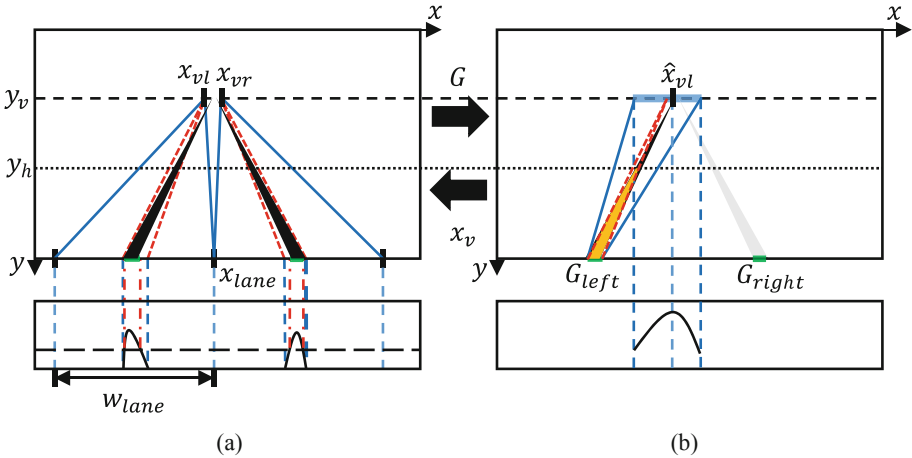


Fig. 3. The diagram of line detections and oriented vanishing points update. (a) Extract lines by histogram of oriented vanishing points. (b) Update oriented vanishing points according to the overlap of lines and lane features. (Color figure online)

The histogram is constructed for each column as shown in Eq. 2:

$$hist_x = \begin{cases} \sum_{h=y_h}^{y_b} I_{bin}(h, x + b_{x_v-x, y_b-y_h}(h-y_h)), & |x - x_{lane}| < w_{lane} \\ 0, & otherwise \end{cases} \quad (2)$$

where

$$x_v = \begin{cases} x_{vl}, x \leq x_{lane} \\ x_{vr}, x \geq x_{lane} \end{cases} \quad (3)$$

and the function $b_{D,H}$ is computed in advance to improve computational efficiency which is a $dy \rightarrow dx$ Bresenham's line mapping space where D, H mean the width and height of Bresenham's line [14]. The required line set G is selected by Eq. 4 where $\lambda \in (0, 1)$ is a proportional coefficient, and both ends of the line are (x_b, y_b) and (x_v, y_v) . Using x_b to represent the line. The result is shown in Fig. 3(a), the green blocks indicates the selected lines.

$$x_b \in G, \text{ if } hist_x > \lambda(y_b - y_h) \quad (4)$$

The lines in G go through lots of lane markings, the x_b of most of them is located in the actual lane lines. Then, we will update the oriented vanishing points through x_b of line set G .

$$O = \sum (I_{bin}(r, c) \wedge I_{line}(r, c)) \quad (5)$$

The image I_{line} is drawn by connecting all the x_b of G_{left} to x_{vl} , as shown in Fig. 3 (b). Define overlap degree O as shown in Eq. 5. O will become larger as the x_v approaches the actual oriented vanishing points. On the basis, we search for oriented vanishing points in the form of regional gradient search. First calculate the maximum overlap O_{mleft} , O_{mright} under a s width on both sides of x_{vl} . Take the maximum value of $O_{x_{vl}}$, O_{mleft} , O_{mright} as O_m . If the maximum position is the boundary of the largest side, continue to search for the s width on the largest side until the maximum value is within the width s , then the new x_{vl} is obtained. Same for x_{vr} .

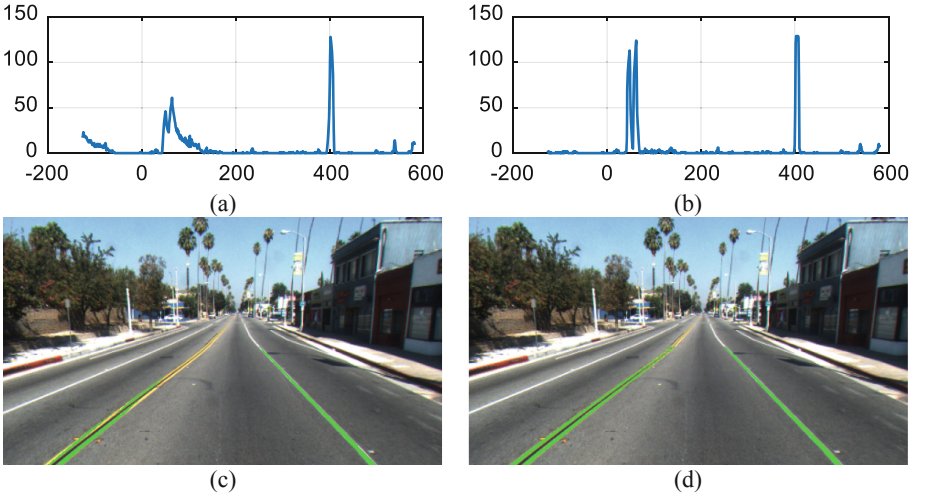


Fig. 4. Comparison of results before and after process. (a) and (b) are origin and final results of histogram; (c) and (d) are origin and final overlaps between lines and features.

In the above, there are two steps, one is to obtain straight lines through the oriented vanishing points, and the other is to find the oriented vanishing points through the straight lines. They form a process of mutual feedback. With multiple iterations, the desired oriented vanishing point \hat{x}_v will be stable and obtained, as shown in Fig. 3. As shown in Fig. 4, after iterations, the lines have been refined.

2.3 Candidate Lane Lines

$$M(c_{left}, c_{right}) \in P, \text{ if } \begin{cases} \left| \frac{c_{left} + c_{right}}{2} - x_{lane} \right| \leq |T_{\Delta x_{lane}}| \\ |c_{right} - c_{left} - w_{lane}| \leq |T_{\Delta w_{lane}}| \end{cases} \quad (6)$$

The lines in \hat{G} are classified according to the neighbor relationship, and the center lines C are obtained. If $|c - x_{lane}| \in [\frac{w_{lane}}{4}, \frac{3w_{lane}}{4}]$, the line is considered as a candidate lane line.

The candidate lane lines are paired as M , and make up a pair set P , as shown in Eq. 6 where $T_{\Delta x_{lane}}$ and $T_{\Delta w_{lane}}$ are the thresholds of lane center change and lane width change selected as $\frac{x_{lane}}{4}$ and $\frac{w_{lane}}{4}$ in this paper. If there is no successfully paired left and right candidate lane lines, the lane line which is closest to the ideal lane line $x_{lane} \pm \frac{w_{lane}}{2}$ is reserved, and another lane line is obtained by the lane width w_{lane} . The pairs in P that minimize the lane center and lane width change $\alpha \left| \frac{c_{left} + c_{right}}{2} - x_{lane} \right| + \beta |c_{right} - c_{left} - w_{lane}|$ are selected as the lane lines.

3 Experiment

This section shows the performance of the proposed method. We use C/C++ to implement the proposed method. The algorithm is built on an Intel Core i5 @ 2.7 GHZ based PC with 8G RAM. The approach is tested on image sequences from Caltech Dataset [17]. The dataset contains four clips of urban streets.

Table 1. Results in Caltech Dataset and comparisons

Scene	Aly's method [17]		Niu's method [18]		Our method		
	AR(%)	FP(%)	AR(%)	FN(%)	AR(%)	FN(%)	Sec/frame
Cordova1	97.2	3.0	92.2	5.4	95.4	4.3	0.0037
Cordova2	96.2	38.4	97.7	1.8	96.8	3.6	0.0037
Washington1	96.7	4.7	96.9	2.5	97.6	2.0	0.0036
Washington2	95.1	2.2	98.5	1.7	99.8	0.2	0.0037
Total	96.3	11.6	96.4	2.85	97.4	2.5	0.0037

The performance of the proposed lane detection method is shown in Table 1. And some sample images and comparisons are shown in Fig. 5. Detection results show robustness in presence of shadows, vehicles and pavements and low time cost in about

4 ms per frame while Aly’s method in about 15 ms per frame. When the left and right lane lines are in non-parallel, the proposed method also has great results.

The failure detection occurs at most in the lane width interruption or roadside as shown in Fig. 6. The reason is that when the lane width suddenly changes, the lane line will be ignored. For roadsides, some of them is selected in feature extraction step. As a result, they may get high scores in histogram and be considered as lane lines. In addition, the threshold applying to histogram to select possible lane lines leads to a delay in response to lane lines that appear in the far distance.



Fig. 5. Detection samples in Caltech Dataset. The first row is the origin image. The second row is result of Aly’s method. And the last row is ours, in where the green points are detected lines covered with features and the red lines are selected lane lines. (Color figure online)



Fig. 6. Some failed samples of our method

4 Conclusion

In this paper, a real-time, efficient and robust method for lane detection is proposed. The proposed method is based on the oriented vanishing points. The statistical histogram is constructed by the oriented vanishing points to detect the lane lines, and the oriented vanishing points of the current frame are estimated according to the lane lines. The mutual feedback process can get robust results of oriented vanishing points with shadows or stains. The best straight lines are validated and selected as lane lines. Experimental results show that the proposed method is robust, accurate and has a low requirement for computing capabilities in different scenes which is meet the requirement of automated driving.

So far, the proposed method uses a straight line model to estimate the geometry of the ego-lane. It is suitable for the roads with small curvature, such as highway and freeway driving. To expand it for the complicated road environments, we will extend

the method with a curve or spline model to describe the lanes in a more accurate way in the future work.

Acknowledgement. This research was funded by National Natural Science Foundation of China (41671441, 41531177, U1764262).

References

1. Lee, C., Moon, J.-H.: Robust lane detection and tracking for real-time applications. *IEEE Trans. Intell. Transp. Syst.* **19**, 4043–4048 (2018)
2. Veit, T., Tarel, J.-P., Nicolle, P., Charbonnier, P.: Evaluation of Road Marking Feature Extraction. In: 2008 11th International IEEE Conference on Intelligent Transportation Systems, pp. 174–181 (2008)
3. Satzoda, R.K., Trivedi, M.M.: Selective salient feature based lane analysis. In: 16th International IEEE Conference on Intelligent Transportation Systems, ITSC 2013, pp. 1906–1911 (2013)
4. Borkar, A., Hayes, M., Smith, M.T., Pankanti, S.: A layered approach to robust lane detection at night. In: 2009 IEEE Workshop on Computational Intelligence in Vehicles and Vehicular Systems, pp. 51–57 (2009)
5. Xiao, J., Luo, L., Yao, Y., Zou, W., Klette, R.: Lane detection based on road module and extended Kalman filter. In: Paul, M., Hitoshi, C., Huang, Q. (eds.) *PSIVT 2017*. LNCS, vol. 10749, pp. 382–395. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-75786-5_31
6. Xiao, J., Tian, H., Zhang, Y., Zhou, Y., Lei, J.: Blind video denoising via texture-aware noise estimation. *Comput. Vis. Image Underst.* **169**, 1–13 (2018)
7. Ozgunalp, U.: Robust lane-detection algorithm based on improved symmetrical local threshold for feature extraction and inverse perspective mapping. *IET Image Proc.* **13**, 975–982 (2019)
8. Nieto, M., Cortés, A., Otaegui, O., Arróspide, J., Salgado, L.: Real-time lane tracking using Rao-Blackwellized particle filter. *J. Real-Time Image Proc.* **11**, 179–191 (2016)
9. Revilloud, M., Gruyer, D., Pollard, E.: An improved approach for robust road marking detection and tracking applied to multi-lane estimation. In: 2013 IEEE Intelligent Vehicles Symposium (IV), pp. 783–790 (2013)
10. Huang, Y., Li, Y., Hu, X., Ci, W.: Lane detection based on inverse perspective transformation and Kalman filter. *KSII Trans. Internet Inf. Syst.* **12**, 643–661 (2018)
11. Lee, Y., Kim, H.: Real-time lane detection and departure warning system on embedded platform. In: 2016 IEEE 6th International Conference on Consumer Electronics – Berlin, ICCE-Berlin, pp. 1–4 (2016)
12. Lotfy, O.G., et al.: Lane departure warning tracking system based on score mechanism. In: 2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS), pp. 1–4 (2016)
13. Zhang, D., Fang, B., Yang, W., Luo, X., Tang, Y.: Robust inverse perspective mapping based on vanishing point. In: Proceedings 2014 IEEE International Conference on Security, Pattern Analysis, and Cybernetics (SPAC), pp. 458–463 (2014)
14. Yoo, J.H., Lee, S., Park, S., Kim, D.H.: A robust lane detection method based on vanishing point estimation using the relevance of line segments. *IEEE Trans. Intell. Transp. Syst.* **18**, 3254–3266 (2017)

15. Hou, C., Hou, J., Yu, C.: An efficient lane markings detection and tracking method based on vanishing point constraints. In: 2016 35th Chinese Control Conference (CCC), pp. 6999–7004 (2016)
16. Li, Q., Zhou, J., Li, B., Guo, Y., Xiao, J.: Robust lane-detection method for low-speed environments. *Sensors* **18**, 4274 (2018)
17. Aly, M.: Real time detection of lane markers in urban streets. In: 2008 IEEE Intelligent Vehicles Symposium, pp. 7–12 (2008)
18. Niu, J., Lu, J., Xu, M., Lv, P., Zhao, X.: Robust lane detection using two-stage feature extraction with curve fitting. *Pattern Recogn.* **59**, 225–233 (2016)



Bypass Enhancement RGB Stream Model for Pedestrian Action Recognition of Autonomous Vehicles

Dong Cao^(✉) and Lisha Xu

Institute of Cognitive Intelligence, DeepBlue Academy of Sciences,
DeepBlue Technology (Shanghai) Co. Ltd., No. 369, Weining Road,
Shanghai, China

doocao@gmail.com, xuls@deepblueai.com

Abstract. Pedestrian action recognition and intention prediction is one of the core issues in the field of autonomous driving. In this research field, action recognition is one of the key technologies. A large number of scholars have done a lot of works to improve the accuracy of the algorithm for the task. However, there are relatively few studies and improvements in the computational complexity of algorithms and system real-time. In the autonomous driving application scenario, the real-time performance and ultra-low latency of the algorithm are extremely important evaluation indicators, which are directly related to the availability and safety of the autonomous driving system. To this end, we construct a bypass enhanced RGB flow model, which combines the previous two-stream algorithm to extract RGB feature information and motion feature information respectively. In the training phase, the two streams are merged by distillation method, and the bypass enhancement is combined in the inference phase to ensure accuracy. The real-time behavior of the action recognition algorithm is significantly improved on the premise that the accuracy does not decrease. Experiments confirm the superiority and effectiveness of our algorithm.

Keywords: Pedestrian action recognition · Autonomous driving · Bypass enhanced

1 Introduction

In the field of autonomous vehicles, pedestrian action recognition and intention prediction is one of the core issues to be solved urgently, which directly affects the process toward a higher level for autonomous vehicles. In the field of pedestrian action recognition and intention prediction, we can start from a variety of perspectives. And behavior recognition is an important part. This paper studies how to improve the real-time behavior recognition algorithm under the premise of ensuring accuracy.

Visual understanding is one of the core issues of artificial intelligence, and it has been rapidly developed with the favorable promotion of deep learning technology. As an important direction of visual understanding, action recognition is the basic work for further application. Current methods to deal with the problem almost following three

ways: (a) Two-stream frameworks that consider spatial and temporal information by taking RGB and optical flow as input [1]; (b) 3D Network that use 3D convolution kernel to extract the spatial and temporal features simultaneously [2]; (c) CNN+RNN that process the visual input with a CNN whose outputs are fed into a stack of RNN (LSTM is common) [3].

These approaches above all prove that the motion information in video plays an essential role in action recognition. As a typical representation of motion information, optical flow is calculated as a drift in a short time, meaning the moment velocity. Moreover, 3D spatiotemporal CNN also found that RGB + optical flow boost their accuracy, and achieve the state-of-the-art result [4] in UCF101 [5] and HM51 datasets [6].

There have been some attempts to describe the optical flow. Dense trajectories track the feature points of frames based on displacement information from optical flow fields, then train the classifiers with the encoded motion features extracted from trajectories [7]. IDT improves the optical flow image by eliminating the camera motion and optimizing the normalization, showing a superior stability but unsatisfactory speed [8]. TV-L1 method is appealing for its relatively good balance between accuracy and efficiency, which iteratively calculates the displacements [9].

Since optical algorithms mentioned above are offline, Flownet was proposed. Flownet is end-to-end trainable, including a generic architecture and a special architecture that contains a layer correlated feature vectors at different locations of image, enabling the online predication of optical flow [10]. Regarding to the quality of flow, Flownet2.0 fuses a stacked network with small displacement network in an optical manner, resulting in a great balance between accuracy and speed on real-world datasets [11]. However, although they were superior in terms of the accuracy, they suffered from extremely expensive computation about time and storage.

In this study, we propose a novel architecture Bypass Enhancement RGB Stream Model. This model leverages the prior information of complex model to obtain the model parameters of motion information during training, and processes RGB information through extended branches in the backbone. Moreover, the model reduces the high computational consumption caused by optical flow, dynamically adjusts the ratio of RGB information to motion information, and avoids over-reliance on optical flow information when using the same ratio to process different dynamic video such as the traditional two-stream model to infer static videos.

2 Related Work

Hallucination. Since the computation of optical flow is time consuming and storage demanding, some attempts to learn other way to replace the flow to represent motion information. [12] proposed that compressed video algorithms can decrease the redundant information, so that can be used accumulated motion vector and residuals to describe motion. Compared to traditional flow methods, motion vectors bring more than 20 times acceleration although a significant drop in accuracy. Some methods represent motion information only by RGB frame [13, 14]. [15] considered that a static image can produce fake motion, thus predict optical flow fields through pre-trained

Im2Flow network. [16] hallucinated optical flow images from videos. Monet models the temporal relationship of appearance features and concurrently calculates the contextual relationship of optical flow features using LSTM [17]. All these approaches describe motion information implicitly but achieve good performance in action recognition.

Generalized Distillation. As a method of model compression [18], knowledge distillation can leverage a more complex model that has been trained to guide a lighter model training, so as to keep the original accuracy of large model while reducing the model size and computing resources [19, 20]. [20] minimized the linear combination of loss functions that are cross entropy with the soft targets and cross entropy with the correct labels. Human society have proved experimentally that interactions between teacher and student can significantly accelerate the learning process. According to this theory, [21] considered a model that supplies privileged information by paradigms when training. [22] applied the privileged information to complex model during distillation to finish knowledge transfer. Recently, for its superior performance in some supervised, semi-supervised and multitask learning tasks, more works derived from distillation and privileged knowledge look forward to improve their tasks [23–27].

3 Our Approach

Two methods were proposed in literature [28], namely MERS and MARS. MARS realizes the information fusion between the optical flow branch and the RGB branch by constructing a two-part loss function. In the phase of model training, it is still necessary to calculate the optical flow and extract the optical flow feature, and then realize feature information transmission from the optical flow feature to the RGB feature branch through the distillation method. In the inference phase, we only use the RGB model to complete the recognition task. It is not necessary to use the optical flow branch, avoiding the calculation of the optical flow. Inference phase can significantly reduce the computational complexity and improve the real-time performance. However, it is at the expense of proper sacrifice accuracy.

In order to solve the problem of this method in [28], we construct a Bypass Enhancement RGB Stream (BERS) Model. It is hoped that the performance of accuracy can be ensured under the condition that the computational complexity is reduced, and the model can infer in real time.

The BERS model structure is shown in Fig. 1. The performer of the model is divided into two different operating states: training mode and inference mode. The overall structure is inspired by [28]. The overall structure in Fig. 1 includes upper and bottom parts. In the training mode, the RGB frames of the video are input into model. RGB information is first sent to the bottom model, and the optical flow frames are processed by the optical flow algorithm module, and then the flow-based action recognition model training is performed. Training is completed to obtain the learned feature weights. Then the global model training is performed, the input is still the RGB frames of the video, and the bottom feature weights are used for the distillation algorithm to assist in learning of the upper model. In the inference mode, the input is

the RGB frames of the video, and only the upper model is activated to implement the inference, and the bottom model does not work. The detailed description is as follows:

Training Phase

In the training mode, the bottom model and the upper model shown in Fig. 1 are all involved in the operation, but the participating operations are sequential and not involved simultaneously.

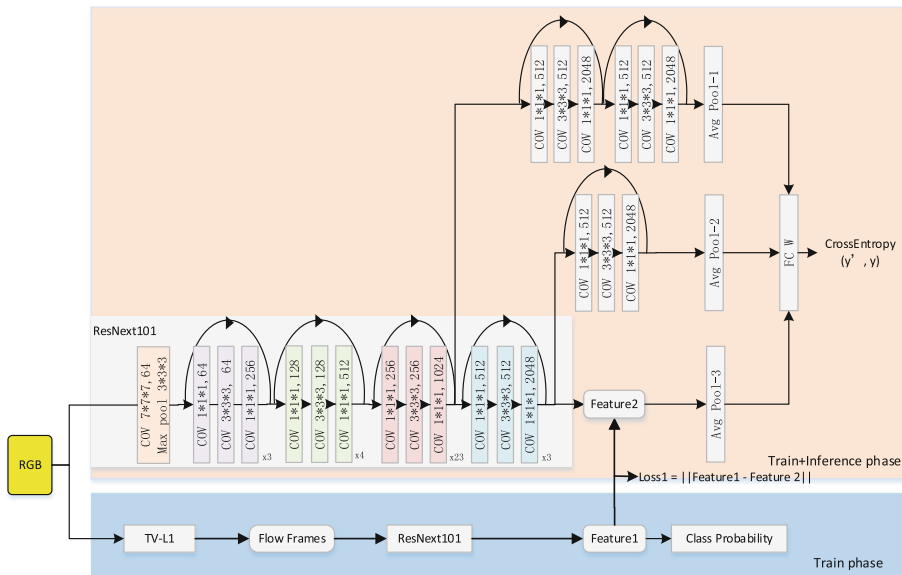


Fig. 1. Bypass Enhancement RGB Stream Model. The bottom is the optical flow branch in training phase, the upper is the enhancement RGB branch.

The first stage is the training of the bottom model. The role of the bottom model is to extract and learn the motion information of the video. The input is RGB frames, which pass through the optical flow algorithm module. Here we use the TV-L1 algorithm to obtain optical flow frames. The optical flow frames are further sent to the deep convolution neural network for motion feature information extraction. We use the network 3D ResNeXt101 [30, 31], the output Feature1 of which is connected to class probability. Then the loss function is constructed according to the cross entropy of class probability and the label y , optimized iteratively to completes the model parameter learning. After the training of bottom model is completed, the valuable material we need to use is the optical flow feature Feature1.

Next, it is the training of the upper model. In this training phase, we need to assist with the pre-trained bottom model. Specifically, the RGB frames are simultaneously input to the bottom model and the upper model, respectively. After the RGB frames enter the pre-trained model in bottom, the optical flow feature Feature1 corresponding to the adjacent RGB frames can be obtained since the parameters of the bottom model

have been fixed. At the same time, the RGB frames are input into the upper model, first enter the backbone neural network to extract the appearance information, and then divided into three branches to construct the first part of the loss function.

The first branch is output to a small residual network-1, then to average pool layer-1, where the input of the first branch is taken from the penultimate ResNext block of deep network 3D ResNeXt101. The second branch is output to another small residual network-2, then to average pool layer-2, where the input point of the second branch is taken between the last convolution layer of the 3D ResNeXt101 and Feature2. The third branch is output to Feature2, then to average pool layer 3. Finally, we combine the above three outputs from Avg pool-1, Avg pool-2, and Avg pool-3 through the fully connected layer, and construct the first part of the loss function with the label y .

$$La = CrossEntropy[W(Avgpoll - 1, Avgpoll - 2, Avgpoll - 3), y] \quad (1)$$

Where W stands for the weights of the fully connected layer.

The second part of the loss function adopts a similar method in [28], and uses the distillation method to realize the transmission of optical flow information to the upper model, called Loss 1. The overall form of the loss function is:

$$L = La + Loss1 = La + \lambda ||Feature1 - Feature2|| \quad (2)$$

Where λ is a hyper-parameter that used to adjust the effect of distillation.

The training of the upper model is completed with the loss function L .

Inference Phase

In the inference mode, the upper model is used while the bottom model does not work. After the original videos are given, the model outputs the action category.

4 Experiment

Dataset. In this section, we will investigate the performance of the Bypass Enhancement RGB Stream framework on datasets Kinetics400 [29] and UCF101 [5]. Kinetics400 consists of 400 classes, including 246 k training videos, 20 k validation videos, and each video are from YouTube, about 10 s. Kinetics is a large-scale dataset whose role in video understanding is similar to ImageNet’s role in image recognition. Some works also migrate to other video datasets using the Kinetics pre-trained model. In this paper, we use the validation dataset to test our trained model. UCF101 is a dataset containing 101 classes that belong to five categories: makeup brushing, crawling, haircutting, playing instruments, and sports. 25 people perform each type of action. Following setting of [28], we use the split1 during training and average 3 splits when testing.

Implementation Details. The novel architecture is composed with source optical flow branch and target derived RGB branch. According to the spirit of distillation and privileged knowledge, when training, we use TV-L1 method [9] to extract optical frames, and save them in jpg format as the input for complex model. Due to the performance of ResNext101 architecture [30, 31], we adopt it to extract the features after inputting the RGB frames and optical flow frames. For the derived branch of

RGB, we choose improved ResNext. Following the setting of [11, 28], we train the model with the SGD optimization method, and use 64 frames clips both in training and testing. As for the weights of loss, grid-search is applied to find the important hyper-parameters. We train the model on Kinetics from scratch while finetune from the pre-trained Kinetics400 model on UCF101.

Results. This part illustrates the superiority of our model from three perspectives. First we compare our model with some single stream models and two-stream models. According to Table 1, we can see that motion information is more accurate than RGB information basically. And two-stream models perform better than single ones. Bypass Enhancement RGB Stream Model is 3.5% higher than MARS while 0.9% lower than MARS+RGB on Kinetics dataset, and 0.9% higher than MARS while 0.1% lower than MARS+RGB on UCF101 dataset. Second in the case of videos that recognize static actions, our model stand out (see Table 2). Third we compare our model with some state-of-the-art models (see Table 3). Apparently our model maintains a good accuracy while reducing a large amount of computing resources.

Table 1. Single-stream model and two-stream models (Dataset: validation of Kinetics and split 1 of UCF101).

Stream	Kinetics	UCF101-1
RGB	68.2%	91.7%
Flow	54.0%	92.5%
MERS	54.3%	93.4%
MARS	65.2%	94.6%
RGB+Flow	69.1%	95.6%
MERS+RGB	68.3%	95.6%
MARS+RGB	69.6%	95.6%
OUR	68.7%	95.5%

Table 2. Recognition on video cases with static actions

Action	MARS	OUR
Making sushi	24%	35.2%
Eating cake	4%	14.1%
Reading newspaper	6%	17.7%

Table 3. Compare with state-of-the-art models (Dataset: validation of Kinetics and average 3 splits of UCF101)

Model	Kinetics	UCF101
Two-stream	69.1%	88.0%
ResNext101	65.1%	94.5%
I3D	71.1%	98.0%
MARS+RGB+FLOW	74.9%	98.1%
OUR	68.7%	97.2%

5 Conclusion

In this paper, we propose a novel model, named Bypass Enhancement RGB Stream Model, to distill the motion information from a complex model during training, avoid expensive computation by only taking RGB images as input when testing. This model combine the appearance features and motion feature through a linear combination of losses, resulting a good balance of accuracy and time in dataset Kinetics and UCF101.

References

1. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: NIPS (2014)
2. Ji, S., Xu, W., Yang, M., Yu, K.: 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(1), 221–231 (2013)
3. Donahue, J., et al.: Long-term recurrent convolutional networks for visual recognition and description. In: CVPR (2015)
4. Carreira, J., Zisserman, A.: Quo vadis, action recognition? A new model and the kinetics dataset. In: CVPR (2017)
5. Soomro, K., Zamir, A.R., Shah, M.: UCF101: a dataset of 101 human actions classes from videos in the wild. *Computer Science* (2012)
6. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: ICCV (2011)
7. Wang, H., Kläser, A., Schmid, C., Liu, C.: Action recognition by dense trajectories. In: CVPR (2011)
8. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: ICCV (2013)
9. Zach, C., Pock, T., Bischof, H.: A duality based approach for realtime TV- L^1 optical flow. In: Hamprecht, F.A., Schnörr, C., Jähne, B. (eds.) DAGM 2007. LNCS, vol. 4713, pp. 214–223. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74936-3_22
10. Dosovitskiy, A., et al.: FlowNet: learning optical flow with convolutional networks. In: ICCV (2015)
11. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: FlowNet 2.0: evolution of optical flow estimation with deep networks. In: CVPR (2017)
12. Wu, C.Y., Zaheer, M., Hu, H., Manmatha, R., Smola, A.J., Krhenbhl, P.: Compressed video action recognition. [arXiv:1712.00636](https://arxiv.org/abs/1712.00636) (2017)
13. Golland, P., Bruckstein, A.M.: Motion from color. *Comput. Vis. Image Underst.* **68**, 356–362 (1997)
14. Singla, N.: Motion detection based on frame difference method. *Int. J. Inf. Comput. Technol.* **4**, 1559–1565 (2014)
15. Gao, R., Xiong, B., Grauman, K.: Im2Flow: motion hallucination from static images for action recognition. In: CVPR (2018)
16. Zhu, Y., Lan, Z., Newsam, S., Hauptmann, A.: Hidden two-stream convolutional networks for action recognition. In: Jawahar, C.V., Li, H., Mori, G., Schindler, K. (eds.) ACCV 2018. LNCS, vol. 11363, pp. 363–378. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-20893-6_23
17. Tang, Y., Ma, L., Zhou, L.: Hallucinating optical flow features for video classification. In: IJCAI (2019)
18. Bucila, C., Caruana, R., Niculescu-Mizil, A.: Model compression. In: ACM SIGKDD (2006)

19. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. In: NIPS Workshop (2015)
20. Papamakarios, G.: Distilling model knowledge. *Comput. Sci.* **14**(7), 38–39 (2015)
21. Vapnik, V., Izmailov, R.: Learning using privileged information: similarity control and knowledge transfer. *JMIR.org* **16**(1), 2023–2049 (2015)
22. Lopez-Paz, D., Bottou, L., Schölkopf, B., Vapnik, V.: Unifying distillation and privileged information. *Computer Science* (2015)
23. Lee, S.H., Kim, D.H., Song, B.C.: Self-supervised knowledge distillation using singular value decomposition. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) *ECCV 2018*. LNCS, vol. 11210, pp. 339–354. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01231-1_21
24. Liu, Y., Sheng, L., Shao, J., Yan, J., Xiang, S., Pan, C.: Multi-label image classification via knowledge distillation from weakly-supervised detection. In: 2018 ACM Multimedia Conference on Multimedia Conference. ACM (2018)
25. Lee, S., Song, B.: Graph-based knowledge distillation by multi-head attention network. arXiv preprint [arXiv:1907.02226](https://arxiv.org/abs/1907.02226) (2019)
26. Liu, Y., Chen, K., Liu, C., Qin, Z., Luo, Z., Wang, J.: Structured knowledge distillation for semantic segmentation. In: CVPR (2019)
27. Park, W., Kim, D., Lu, Y., Cho, M.: Relational knowledge distillation. In: CVPR (2019)
28. Crasto, N., Weinzaepfel, P., Alahari, K., Schmid, C.: MARS: motion-augmented RGB stream for action recognition. In: CVPR (2019)
29. Kay, W., et al.: The kinetics human action video dataset. arXiv preprint [arXiv:1705.06950](https://arxiv.org/abs/1705.06950) (2017)
30. Hara, K., Kataoka, H., Satoh, Y.: Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and ImageNet. In: CVPR (2018)
31. Xie, S., Girshick, R., Dollár, Piotr, Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: CVPR (2017)



Paved and Unpaved Road Segmentation Using Deep Neural Network

Dabeen Lee, Seunghyun Kim, Hongjun Lee, Chung Choo Chung,
and Whoi-Yul Kim^(✉)

Hanyang University, 222 Wangsimri-ro, Seoul 04763, South Korea
{dblee, shkim, hongjunlee}@vision.hanyang.ac.kr,
{cchung, wykim}@hanyang.ac.kr

Abstract. Semantic segmentation is essential for autonomous driving, which classifies roads and other objects in the image and provides pixel-level information. For high quality autonomous driving, it is necessary to consider the driving environment of the vehicle, and the vehicle speed should be controlled according to types of road. For this purpose, the semantic segmentation module has to classify types of road. However, current public datasets do not provide annotation data for these road types. In this paper, we propose a method to train the semantic segmentation model for classifying road types. We analyzed the problems that can occur when using a public dataset like KITTI or Cityscapes for training, and used Mapillary Vistas data as training data to get generalized performance. In addition, we use focal loss and over-sampling techniques to alleviate the class imbalance problem caused by relatively small class data.

Keywords: Semantic segmentation · Class imbalance · Road type · Autonomous driving

1 Introduction

Several global companies, including General Motors (GM), Audi, Google and Tesla, are involved in the development of autonomous vehicles. In particular, the GM Cadillac CT6 is equipped with the ‘Super Cruise’ feature which is level 3 autonomous driving. While using the feature, the driver does not have to hold the steering wheel. Because these kinds of autonomous driving features can provide convenience to customers, autonomous driving technology is rapidly developed and commercialized. However, since the malfunction of autonomous vehicles is a direct threat to human safety, the autonomous driving requires to accurately recognize road and objects that the vehicle have to avoid, such as people. To do this, when using the camera-based autonomous driving feature, the boundary of objects in the image should be clearly distinguished. Object detection based on bounding box such as YOLO [1], can provide object class and location information, however the location is not highly accurate because the result is displayed as bounding box. Semantic segmentation, on the other hand, is pixel-level image classification that can be clearly distinguish the boundaries of objects. This is the reason that semantic segmentation is more appropriate to autonomous driving than object detection. KITTI [2] and Cityscapes [3], which are

representative of public driving datasets, provide ground truth (GT) for semantic segmentation training and promote development of semantic segmentation model by benchmark competition.

When driving a car, the driver controls the vehicle speed depending on type of driving road for safety and riding comfort. Traffic law regulates maximum speed of paved and unpaved roads differently. If you drive unpaved road like driving on paved road, accidents such as vehicle rollover can occur and damage to human and property. Therefore, autonomous driving feature should classify road types and control vehicle speed according to road types for safety. There have been several related works to predict road types or road condition. There is a work [4] to estimate the road type by calculating shear strength with physical quantities such as friction coefficients and slip angles acquired from on-board rover sensors. Since the method only measure the physical quantities to the contact area between the ground and tire, the road surface condition cannot be predicted before the vehicle passes. Wang *et al.* [5] proposed a method to estimate road types by using LiDAR sensor. They reconstruct three dimensional road surface by using LiDAR data and extract features from road surface. And then, they classify road types with Support Vector Machines by using extracted features. However, due to the characteristics of LiDAR sensor, it cannot acquire dense information for the road surface and sometimes fail to collect data for the highly reflective surface. LiDAR also has a disadvantage that the cost is much higher than a camera. There have been several works using cameras [6, 7]. Slavkovikj *et al.* [6] proposed a patch-wise road type classification method using K-means clustering. This method can classify paved and unpaved road, but it cannot provide a pixel-wise classification results and has the limitation that additional method is needed to distinguish road and non-road region. Roychowdhury *et al.* [7] proposed a method to estimate road friction by predicting road condition such as dry, wet, snow and icy. The method considers the front area of the vehicle on the image as road surface. Then, the road surface area is divided into 5 by 3 patches. A three-layer convolutional neural network is applied to each patch to recognize the type of road surface. However, this method cannot guarantee that the patch is a real road surface in situations like curves or corners.

This paper proposes a method to training semantic segmentation model by using modified public dataset to distinguish paved and unpaved road from driving images. Typically, deep learning-based semantic segmentation module requires a large dataset to provide pixel-wise annotation of GT for training. For this reason, we use public driving datasets to save effort and cost in creating such a large dataset. We compared three public datasets, KITTI, Cityscapes, and Mapillary Vistas [8], as candidates for training datasets, and decided to use Mapillary Vistas which provides various road scenes. Since the dataset does not categorize roads by type, we subdivide the ‘roads’ label into ‘paved roads’ and ‘unpaved roads’. As a result, we confirmed that the number of data of paved road class and unpaved road class is imbalance. To alleviate this problem, over-sampling and focal loss [9] are applied.

The paper is organized as follows. In Sect. 2, we describe semantic segmentation model, datasets and methods that used to alleviate class imbalance problem. In Sect. 3, we present experimental results. Finally, conclusions and future direction of the research are given in Sect. 4.

2 Proposed Method

In this section, we introduce the semantic segmentation model used to classify road types and the reasons for the selection. To determine appropriate dataset from public dataset, each datasets is analyzed and experiments are performed. After modifying the dataset, the ‘Road’ class is subdivided into ‘Paved road’ and ‘Unpaved road’, and it cause class imbalance problem due to the small amount of ‘unpaved roads’ dataset. We apply the over-sampling and focal loss to improve segmentation performance.

2.1 Semantic Segmentation Model

In this paper, we use AdapNet [10] as a segmentation module, which shows good performance in the Cityscapes benchmark. The AdapNet consists of an encoder part that contracts a segment and a decoder part that expands a segment. The AdapNet adopt ResNet-50 [11] as an encoder to create a deeper network and proposed multiscale blocks to reduce computational cost and dilated-convolution to acquire high resolution features. There are some models outperform AdapNet, such as DeepLab [12], but these models require a lot of memory, making them difficult to train on a single GPU. Because deep learning systems installed in commercial vehicles are less cost effective if they require multi-GPU platform, we choose AdapNet as a segmentation model, which can be trained on a single GPU and showed high performance on the Cityscapes benchmark.

2.2 Dataset

Training Dataset. Three candidate datasets were considered for training the segmentation module: KITTI, Cityscapes and Mapillary Vistas. KITTI is a dataset consisting of 200 images of 1242×375 resolution. Cityscapes is a dataset consisting of approximately 5,000 images of 2048×1024 resolution and all images have the same viewpoint as shown in Fig. 1(a) because all images were captured by a camera installed on the same vehicle. Mapillary Vistas is a dataset created by collecting approximately 25,000 full HD road images of at least 1920×1080 resolution. Mapillary Vistas images are captured from various types of vehicles, people, or CCTVs. Therefore, they have various viewpoints as shown in Fig. 1(b). We exclude the KITTI dataset because the number of data is too small to be used for training.

The remaining two datasets, Cityscapes and Mapillary Vistas, have following characteristics: The Cityscapes dataset has the same viewpoint, and Mapillary Vistas has different viewpoints. To determine which dataset is suitable for training, we train the semantic segmentation module with the two datasets and the results are compared. We have integrated objects labels into ‘Background’ label except ‘Road’ because the labels provided by Cityscapes and Mapillary Vistas do not exactly match. Therefore, we made datasets which consists of two classes, road and background. After refining process, excluding images that do not include roads in the image, the training set of the Cityscapes is 2,907 images and the Mapillary Vistas is 11,884 images. The excluded image examples for Cityscapes and Mapillary Vistas are shown in Fig. 2(a) and (b),

respectively. For the comparison, we randomly select 2,907 images from Mapillary Vistas dataset equal to the number of Cityscapes and use them as training datasets.



Fig. 1. Dataset example images: (a) Example images of the Cityscapes dataset. (b) Example images of the Mapillary Vistas dataset. (c) Example images of the PG dataset.



Fig. 2. Excluded images from training: (a) Excluded images from Cityscapes. (b) Excluded images from Mapillary Vistas.

Three test datasets were used for performance evaluation after the training. The first is refined Cityscapes test dataset consisting of 492 images. The second was a road driving dataset taken by us. We took images with a camera mounted on a Hyundai Tucson ix2WD 2013 model during driving on a proving ground (PG) road. As with the Cityscapes dataset, all images have the same viewpoint but the camera pose is different from the Cityscapes data. PG dataset consists of 391 images of 2048×1536 resolution. The last one is the Mapillary Vistas dataset. Since the Mapillary Vistas dataset does not provide validation data, we randomly extracted 500 images that were not used as training data.

After training two segmentation modules, one is trained using the Cityscapes training dataset, and the other is trained using the Mapillary Vistas training dataset, segmentation performance was evaluated by calculating mean intersection over union

(mIOU) for road class of test datasets. As shown in Table 1, the module trained with Cityscapes showed good performance on the Cityscapes test dataset, but was poor on the other two test datasets. While, the model trained with Mapillary Vistas showed almost constant generalized performance regardless of the test datasets. The module trained using Cityscapes seems to occur overfitting due to the fixed geometric conditions of the training images. Thus, we select the Mapillary Vistas dataset as the training dataset for the segmentation module.

Table 1. mIOU of the ‘road’ class for each of the training and test datasets

Training dataset	Test dataset		
	Cityscapes	PG	Mapillary Vistas
Cityscapes	98.29%	91.22%	93.63%
Mapillary Vistas	95.54%	98.01%	95.26%

The original Mapillary Vistas has a ‘Road’ class, regardless of the type of road. We subdivide this label into ‘Paved road’ and ‘Unpaved road’ to classify road types. Therefore, the segmentation module outputs three probability map: background, paved road, and unpaved road. As a result of the dataset modification, we got 11,810 paved road class data and 74 unpaved road class data, and we divided them by a ratio of 7:3 for each class to create a training dataset of 8,267 paved and 52 unpaved road data and a test dataset of 3,565 paved and 22 unpaved road data.

Test Dataset. Two test datasets were used to evaluate the trained model. One is aforementioned Mapillary Vistas test dataset. The other is PG dataset captured from Korea Automobile Testing & Research Institute in Hwaseong-si, Republic of Korea. the PG dataset consists of 391 paved and 166 unpaved road data.

2.3 Class Imbalance

The term ‘class imbalance’ means that the number of each class in the dataset is significantly different. Deep learning model can be trained without biasing particular class when the class ratio of the training dataset is balanced. However, if the model is trained with a class imbalanced dataset, the segmentation performance for classes with fewer data is poor. The modified Mapillary Vistas has 8,267 paved road class data and 52 unpaved road class data. The difference in the number of data between the two classes is very serious, resulting in low performance for the unpaved road class. We apply over-sampling and focal loss to alleviate this problem.

Over-Sampling. Over-sampling changes the proportion of classes by copying the data of a small number of classes. In this paper, we attempt to alleviate the class imbalance problem by over-sampling the unpaved road class and find the appropriate ratio between the number of pavement class and the unpaved road class.

Focal Loss. Focal loss is a proposed method to solve the problem that the background rate of object candidates is much larger than objects in the field of 1stage object

detection. Focal loss improves the training contribution of features that are difficult to classify by reducing the loss of features which are easy to classify. Focal loss is a formula obtained by adding scaling factor to cross-entropy loss. If the predicted probability for any t class is p_t , we define the focal loss as:

$$FL(p_t) = -\alpha_t(1 - p_t)^\gamma \log(p_t). \quad (1)$$

The training contribution can be controlled by changing the values of α_t and γ . In this paper, we tried to mitigate the class imbalance problem by using focal loss in the training process, focusing on the unpaved road class rather than the background and the paved road class, which are relatively easy to training.

3 Experiment Results

We implemented AdapNet in Python Tensorflow environment and trained segmentation module for classifying road types. The parameters are set as follows: input image size is 768×384 , batch size is 8, max iteration is $1.5 * 10^5$, optimizer is Adam optimizer, polynomial decay learning rate is used with $1 * 10^{-4}$ initial learning rate and $1 * 10^{-4}$ power. Focal loss parameter is set to α_t to 0.25 and γ to 2, suggested as the best in the original paper. The training data were over-sampled so that the number of unpaved road images would be 10%, 25%, 50%, and 100% of the number of paved road images. When the over-sampling was not applied, the ratio between the two classes was 0.6%. The trained models were evaluated by calculating the mIOU for each class using the PG test dataset and the Mapillary Vistas test dataset.

Table 2 shows the performance of models that using focal loss for training process and not. The dataset used for training was not over-sampled. In the case of the PG test dataset, the paved road class performance increased by 0.52% when focal loss was used, but the unpaved road class performance decreased by 0.13%. In the case of the Mapillary Vistas test dataset, the unpaved road class performance increased by 20.08% when the focal loss was used, and the paved road class performance decreased by 0.3%. In both cases, the average mIOU is increased, and this result shows that the focal loss is effective to enhance the segmentation performance.

Table 2. Segmentation performance of models using and not using focal loss

Test dataset	Loss	mIOU (Paved)	mIOU (Unpaved)	mIOU (Average)
PG	Cross entropy loss	93.48%	94.66%	94.07%
	Focal loss	94.00%	94.53%	94.26%
Mapillary Vistas	Cross entropy loss	93.15%	24.27%	58.71%
	Focal loss	92.89%	44.35%	68.62%

Table 3 shows the performance of the models trained with datasets which have different class ratio by over-sampling the unpaved road. In the PG test dataset, performance was best when the number of unpaved roads was 25% of the paved road, and best when the class rate was 50% in the Mapillary Vistas test dataset. At 100% class rate dataset which over-sampled too much, both test datasets showed low performance.

Table 3. Segmentation performance of models applied over-sampling

Test dataset	Over-sampling rate	mIOU (Paved)	mIOU (Unpaved)	mIOU (Average)
PG	0.6%	93.48%	94.66%	94.07%
	10%	95.07%	96.66%	95.86%
	25%	95.26%	96.81%	96.03%
	50%	94.27%	96.95%	95.61%
	100%	84.21%	62.29%	73.25%
Mapillary Vistas	0.6%	93.15%	24.27%	58.71%
	10%	93.07%	55.23%	74.15%
	25%	91.78%	63.05%	77.41%
	50%	92.66%	64.91%	78.79%
	100%	93.11%	41.27%	67.19%

Table 4 shows the performance of the models trained with over-sampling datasets and applied focal loss. Applying both over-sampling and focal loss to the training process result in lower performance than just over-sampling alone.

Table 4. Segmentation performance of models applied over-sampling and focal loss

Test dataset	Over-sampling rate	mIOU (Paved)	mIOU (Unpaved)	mIOU (Average)
PG	10%	93.96%	93.96%	93.96%
	25%	93.63%	93.99%	93.81%
	50%	80.06%	73.11%	76.58%
	100%	4.09%	27.92%	16.00%
Mapillary Vistas	10%	92.92%	50.65%	71.79%
	25%	91.68%	51.55%	71.62%
	50%	73.10%	3.27%	38.19%
	100%	5.19%	0.72%	2.96%

The results show that the segmentation performance improved the most when only over-sampling was used. However, depending on the test dataset, the optimal over-sampling ratio changes, and the more over-sampling required the longer training time. It is difficult to determine the best over-sampling ratio simply comparing the evaluation results. On the other hand, focal loss does not require additional training time because

original training dataset is used, but the performance improvement when using focal loss is relatively low compared to when over-sampling is used. In this paper, we use the model trained using only 50% over-sampling as a segmentation module, and Figs. 3 and 4 show the results of predicting the PG test dataset and the Mapillary Vistas test dataset using the segmentation module.

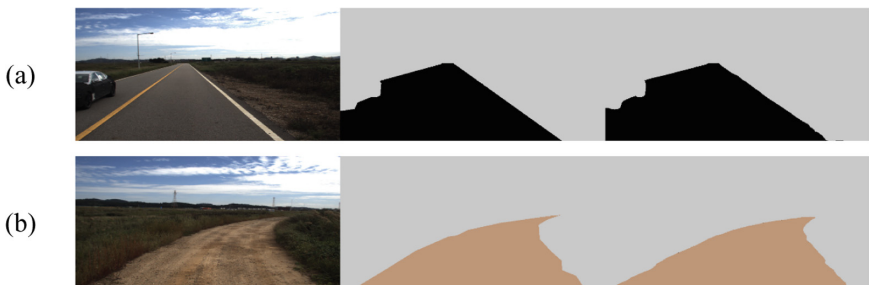


Fig. 3. Prediction result of 50 over-sampling model with PG test dataset, the left column is the original image, the middle column is the GT, and the right column is the prediction: (a) Paved road class prediction. (b) Unpaved road class prediction.



Fig. 4. Prediction result of 50 over-sampling model with Mapillary Vistas test dataset, the left column is the original image, the middle column is the GT, and the right column is the prediction: (a) Paved road prediction result. (b) Unpaved road prediction result.

4 Conclusion

This paper proposed a method for training deep learning based segmentation module that segment paved and unpaved roads for safe autonomous driving. We analyzed two public dataset Cityscapes and Mapillary Vistas. The Cityscapes was taken with the same camera pose, occurring overfitting. To prevent this, Mapillary Vistas which consisting with various viewpoint data was used for training dataset. Since the Mapillary Vistas did not distinguish road labels by road type, we subdivided road labels into ‘Paved road’ and ‘Unpaved road’, and removed images that did not include roads. The modified Mapillary Vistas has very few ‘Unpaved road’ data, and it cause class imbalance problem. To alleviate this problem, we applied over-sampling and focal loss

and confirmed that these techniques improve segmentation performance. As future work, we plan to add other road types, such as ‘brick road’, and to acquire more unpaved road data to improve the performance of the proposed model. Also, we will try to overcome class imbalance problem through structural improvement of the segmentation model.

Acknowledgments. This work was in parts supported by Korea Evaluation Institute of Industrial Technology (KEIT) grant funded by the Korea government (MOTIE) (No. 20000293, Road Surface Condition Detection using Environmental and In-vehicle Sensors).

References

1. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
2. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: the KITTI dataset. *Int. J. Robot. Res.* **32**, 1231–1237 (2013)
3. Cordts, M., et al.: The cityscapes dataset for semantic urban scene understanding. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3213–3223 (2016)
4. Iagnemma, K., Kang, S., Shibly, H., Dubowsky, S.: Online terrain parameter estimation for wheeled mobile robots with application to planetary rovers. *IEEE Trans. Robot.* **20**(5), 921–927 (2004)
5. Wang, S., Kodagoda, S., Ranasinghe, R.: Road terrain type classification based on laser measurement system data. In: Australasian Conference on Robotics and Automation (ACRA) (2012)
6. Slavkovicj, V., Verstockt, S., De Neve, W., Van Hoecke, S., Van de Walle, R.: Image-based road type classification. In: International Conference on Pattern Recognition (CVPR) (2014)
7. Roychowdhury, S., Zhao, M., Wallin, A., Ohlsson, N., Jonasson, M.: Machine learning models for road surface and friction estimation using front-camera images. In: International Joint Conference on Neural Networks (IJCNN) (2018)
8. Neuhold, G., Ollmann, T., Bulow, S.R., Kotschieder, P.: The mapillary vistas dataset for semantic understanding of street scenes. In: The IEEE International Conference on Computer Vision (ICCV), pp. 4990–4999 (2017)
9. Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollar, P.: Focal loss for dense object detection. In: The IEEE International Conference on Computer Vision (ICCV), pp. 2980–2988 (2017)
10. Valada, A., Vertens, J., Dhall, A., Burgard, W.: AdapNet: adaptive semantic segmentation in adverse environmental conditions. In: International Conference on Robotics and Automation (ICRA) (2017)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: International Conference on Pattern Recognition (CVPR) (2015)
12. Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **40**, 834–848 (2018)



Image Scene Conversion Algorithm Based on Generative Adversarial Networks

Honggang Xie^{1(✉)}, Jinsheng Xiao^{2(✉)}, Junfeng Lei², Wenjuan Xie³,
and Reinhard Klette⁴

¹ School of Electrical and Electronic Engineering,
Hubei University of Technology, Wuhan 430068, China
xiehg@hbut.edu.cn

² School of Electronic Information, Wuhan University, Wuhan 430072, China
xiaoj@swhu.edu.cn

³ Fiber Home Telecommunication Technologies Co., Ltd.,
Wuhan 430074, China

⁴ School of Engineering, Computer and Mathematical Sciences,
Auckland University of Technology, Auckland, New Zealand

Abstract. This paper presents an image scene conversion algorithm based on generative adversarial networks (GANs). First, the generator uses the generator network with cross-layer connection structure to realize the sharing of image structure information, so that the structure and edge of the generated image are consistent with the input image as far as possible. Secondly, the multi-scale global convolution network discriminator is used to determine different scales of image. Then, the combinational loss functions including GAN, L1, VGG and feature matching (FM) are designed. The network structure of the generator, the number of multi-scale discriminator and the weighted combination of multiple loss functions are evaluated and analyzed, and the optimized algorithm structure is given. Finally, through image fogging and day-to-night conversion experiment, the results show that the details of the converted image are more complete and the generated image is more realistic.

Keywords: Image conversion · Generative adversarial networks · Deep learning · Image generation

1 Introduction

Computer vision can be thought of as a “translation” input image, and a scene can be represented by a map, a hand-drawn, or a photo. In unsupervised learning, image-to-image conversion problems are difficult to achieve because training images do not match, i.e., are not paired training sets. In the supervised learning, the corresponding images can be trained and matched in different domains [1], which can make the mapping relationship between the generated image and the input image pixels more accurate, and avoid the phenomenon that the generated image is uncontrollable in the unsupervised learning.

The use of convolutional neural networks (CNN) for supervised learning is the research direction of many scholars. When generating images using network prediction, the L1 loss function is often used to calculate the Euclidean distance between the predicted image and the real image, which may produce ambiguous results [2, 3]. GANs [4, 5] uses training to generate models that attempt to determine whether the output image is real or falsified, and its loss function can be applied to traditionally require very different kinds of tasks. How to use the optimized GANs for supervised learning and realize various transformations of images has gradually become a research hotspot. Pix2pix [1] uses the condition GANs [6] for different image conversions. In the absence of training pairs, various methods for image-to-image translation have also been proposed [7]. Chen et al. [8] pointed out that due to training instability and optimization problems, the conditional GANs training is difficult to generate high-resolution images, and perceptual loss [9] is an idea that can solve this problem.

As a representative of the field of image conversion, image style conversion is mainly divided into two categories, one is based on the global mean by matching the mean [10] and variance of the pixel color or its histogram to achieve styling; the other is based on local stylization of images by dense correspondence between content and style photos based on low-level or high-level features [11]. These methods are slow in practice and, in addition, they are usually for specific scenes (for example, day or season changes). Gatys et al. [12] proposed an art-style transformation algorithm. The main step is to solve the problem of extracting deep features from Gram matrix from content image and style image. Although the performance and speed are further improved by algorithm improvement [13, 14]. However, these methods sometimes produce images that are not real enough.

Based on the above analysis, we proposes a novel image scene conversion algorithm. There are three main contributions in this work: (1) a new generator with a cross-layer connection structure is designed, which better preserves the structural information of the image; (2) a multi-scale discriminator is designed, which can take into account the details and structure of the image; (3) a new combined loss function is designed, adding VGG loss and FM loss, and increasing the control of generating against the network.

The remainder of paper is organized as follows. Section 2 analyzes the algorithm in detail from three aspects: generator structure, discriminator structure and loss function. The experimental details and evaluations are presented in Sect. 3. We finally conclude our work in Sect. 4.

2 Image Conversion Algorithm Based on Generative Adversarial Networks

The image scene conversion algorithm based on GANs proposed in this paper is mainly divided into two stages of training and testing. The GANs model is optimized during the training phase so that the input image is obtained through the GANs model during the test phase. Optimize network parameters by iteratively generating networks and decision networks. This section describes the generator, the discriminator and the loss function.

2.1 Generator Structure

This paper uses a cross-layer connection on the generator network G design because there is a large amount of information shared between input and output in image conversion, and it needs to transmit this information directly on the network. For example, when scene conversion is performed, the input and output share the position of the highlighted edge. The network structure is shown in Fig. 1.

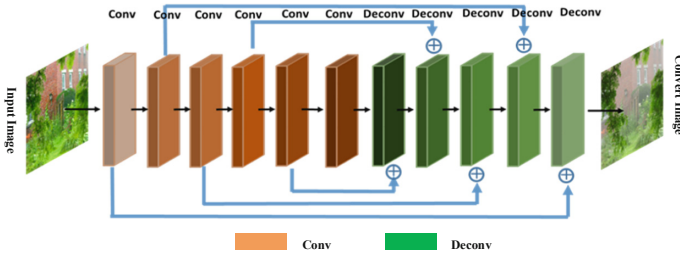


Fig. 1. Generator network structure

As shown in Fig. 1, the network structure is bilaterally symmetric, with the convolution operation on the left and the deconvolution operation on the right. In the Convolution layer, Batch Normalization and Prelu as a module, marked as a layer. The input image is subjected to a multi-layer convolution operation to obtain an intermediate layer. At the same time, the convolutional layer information corresponding to the right side and the left side is directly connected, and finally the output of the image is obtained.

2.2 Discriminator Structure

Multi-scale Discriminator Network

Improving the network's receptive field can use deeper networks or larger convolution kernels, but both increase network capacity and can lead to overfitting. In addition, both of these methods require more memory. So a multi-scale discriminator is used in this paper, which used to determine each of the different scales. For high-resolution images, multi-scale discriminator can improve the network's receptive field.

This paper uses up to three discriminators, which are recorded as D1, D2 and D3. When three discriminators are used, the images are downsampled twice and then judged. A single discriminator downsampling layer and an output decision layer are composed, except that the size of the input image is different.

In theory, the more discriminator, the better, but it is not. First, the more discriminators increase the complexity and computation of the network; second, the number of discriminator is related to the size of the input image itself. If the input size is appropriate, not large or super large, the discriminator does not need to be excessive. Therefore, it is appropriate to choose the number of discriminator, this paper conducts test experiments on the fogging training set. The image input size is 256×256 , when the number of discriminator is 1, 2, and 3, the effect of 60 epoch is iterated, as shown in Fig. 2.

Overall, the number of discriminator has little effect on the content of the generated image. But there will be differences in detail. It can be seen from Fig. 3 when $\text{num_D} = 1$ and $\text{num_D} = 3$, the details of the scene will be missing, such as the horizontal line of the building in the figure, and when num_D is 2, it can be retained. At the same time, in the sky part of the figure, distortion occurs when $\text{num_D} = 1$. When $\text{num_D} = 2$, the sky color is more uniform. Therefore, $\text{num_D} = 2$ for all experiments in this paper.

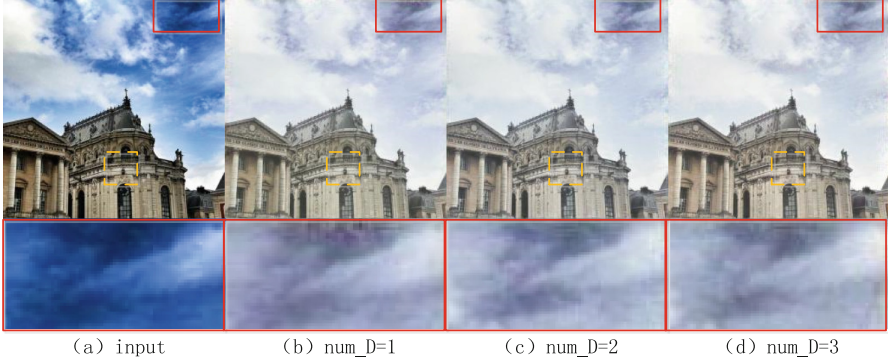


Fig. 2. Comparison of different number of discriminator

2.3 Loss Function

Loss Function Composition

The loss function in this paper consists of four parts, namely GAN loss, L1 loss, VGG loss and FM loss, the ultimate optimization objective of the total loss function of the algorithm in this paper can be expressed as:

$$\min_G \left(\max_{D_1, D_2, D_3} \sum_{k=1,2,3} L_{GAN}(G, D_k) + \lambda_1 L_1(G) + \lambda_2 \sum_{k=1,2,3} L_{FM}(G, D_k) + \lambda_3 L_{VGG}(G) \right) \quad (1)$$

Each loss function is defined as:

$$\left\{ \begin{array}{l} L_{GAN}(G, D_k) = E_{(x,y)} [\log D_k(y)] + E_x [\log(1 - D_k(G(x)))] \\ L_{L1}(G) = E_{(x,y)} [\|y - G(x)\|_1] \\ L_{FM}(G, D_k) = E_{(x,y)} \sum_{i=1}^T \frac{1}{N_i} \left[\left\| D_k^{(i)}(x, y) - D_k^{(i)}(x, G(x)) \right\|_1 \right] \\ L_{VGG}(G) = \sum_{i=1}^N \frac{1}{M_i} \left[\left\| F^{(i)}(y) - F^{(i)}(G(x)) \right\|_1 \right] \end{array} \right. \quad (2)$$

Where x is the input image and y is the target image, T is the total number of layers of the discriminator, and N_i is the number of elements in each layer, $F^{(i)}$ represents the i th layer of VGG network, and M_i represents the number of elements in this layer.

In this paper, the total loss, no VGG loss, no L1 loss and FM loss were tested. The experimental results in the three cases are shown in Fig. 3. When the VGG loss is not used, image distortion occurs. In Fig. 3, the red frame area, in the sky, the track, etc., will appear as an irregular white oval “foreign object”. This situation may occur due to data overflow. When there is no loss of L1 and FM, the image will not be distorted, but the color of the image will be deviated.

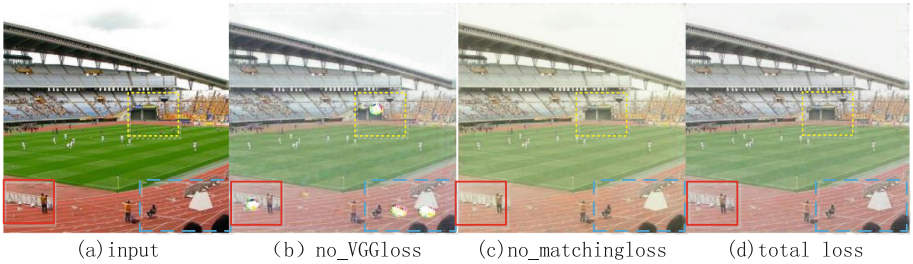


Fig. 3. Comparison of different loss function results (Color figure online)

3 Experimental Results and Discussion

3.1 Experimental Environment

The fogging training dataset [15] uses the software Adobe lightroom CC fogging function to fog the Middlebury Stereo Datasets and the fog-free images collected online. Add fog of 30, 40, 50, 60, 70, 80, 90, 100 to 76 fog-free images, and eventually forming 608 pairs of foggy images with different concentrations of fog. A training set is made with matching image pairs of the fog-free image. There were 17,112 day and night conversion training dataset [16].

3.2 Subjective Results Analysis

Using fogging training dataset, the pix2pix [1], CycleGAN [7], DRPAN [5] algorithm and software fogging scene conversion experiment are compared, and using day and night training dataset, the algorithm of this paper performs scene conversion test. The results are shown in Figs. 4.

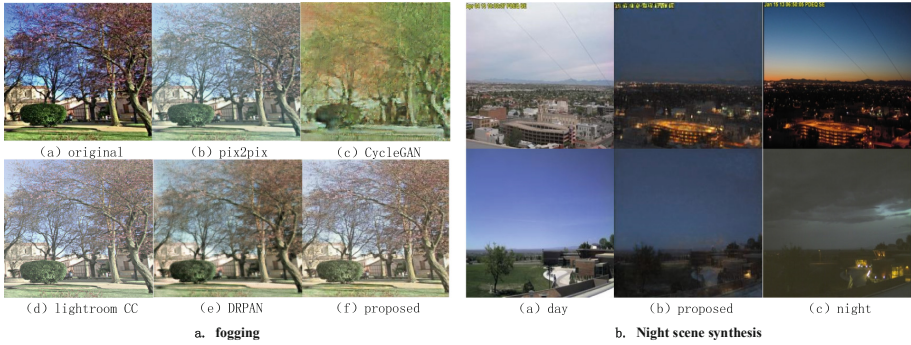


Fig. 4. Comparison of fogging results and night scene synthesis (Color figure online)

It can be seen from Fig. 4a that the image after pix2pix processing has a fogging effect. The content of the image is clear and the details are not lost, but the image is fogged and the overall color of the image is blue. The content of CycleGAN is blurred and the color is seriously distorted. The effect of software fogging is very similar to the proposed method. The fogging is uniform, the color of the fog is not biased, and the details of the image are well preserved. For the DRPAN, although the color of the whole image has decreased, the image is blurred. Especially the trunk and leaves above the image, without borders, which is very blurred.

In Fig. 4b, the content of the image is basically unchanged, and the brightness of the generated night sky area is significantly different from that of the non-sky area. Although it is different from the real night scene, the sky is darkened overall, and the village and buildings are lit, which is more realistic. In summary, when the day-to-night transition is performed, the converted image as a whole exhibits the characteristics of the night, but sometimes an unreal situation occurs.

3.3 Objective Index Analysis

Analysis of Objective Index of Image Fogging

In this paper, the fog concentration (FADE) [17] is used to obtain the fog concentration index, PSNR and SSIM objective indicators for 40 images in the test set. Table 1 shows the mean and mean square error of the three indicators, and compares the fog-free image, CycleGAN, Pix2pix, DRPAN, and software fogging effects.

Table 1. Comparison of objective index of fogging

	Fog-free image	CycleGAN	Pix2pix	DRPAN	Software fogging	Proposed
FADE	0.230 ± 0.116	0.736 ± 0.431	0.689 ± 0.348	0.459 ± 0.250	0.670 ± 0.412	0.634 ± 0.410
PSNR	19.231 ± 2.157	12.703 ± 1.708	13.830 ± 0.690	17.211 ± 2.541	13.611 ± 1.147	14.760 ± 0.733
SSIM	0.789 ± 0.052	0.349 ± 0.084	0.752 ± 0.074	0.809 ± 0.082	0.782 ± 0.075	0.725 ± 0.062

It can be seen from Table 1 that the FADE of the image significantly increased, with the lowest fogging degree compared with DRPAN and the highest degree of CycleGAN. Pix2pix is similar to the algorithm in this paper and the fogging degree of software. The PSNR and SSIM values of the image basically maintain a small fluctuation within a certain range, while DRPAN and CycleGAN have large fluctuations. In addition, the PSNR value of CycleGAN is basically the lowest among several fogging algorithms, which is also due to errors and deficiencies in the image content generated by this algorithm. In contrast, the overall PSNR and SSIM values of DRPAN are relatively high, because the structure content is well preserved, but the fogging effect is not obvious.

Analysis of Objective Index of Day-Night Conversion

In this paper, the image quality after conversion is evaluated by image average brightness (IAB) and image sharpness (IC). The IAB is obtained by reading the Y channel value and performing normalization calculation. The IC is calculated using the Leningrad gradient method. The correlation values for calculating the two sets of images shown in Fig. 4b are shown in Table 2.

Table 2. Comparison of objective index of day-night conversion

	Day-top	Proposed	Night-top	Day-bottom	Proposed	Night-bottom
IAB	154.8823	63.8774	68.4052	114.6836	62.1936	70.8118
IC	4.84964	2.69571	3.37494	2.33589	0.802856	1.17264

After the day-night conversion of the algorithm, the average brightness of the image IAB indicator is significantly smaller than the average brightness of the daytime image, which is similar to the average brightness of the real night image; the IC value of the image is lower than the IC value of the real night scene, maintaining its basic characteristics but blurring the texture details.

4 Conclusions

The content of image scene conversion algorithm based on GANs is introduced. Firstly, the design of cross-layer connection generator network, multi-scale decision-maker network and four combinations of loss function are introduced. Then the performance of the network module is analyzed and the rationality of the algorithm design is proved by experiments. Then it introduces the software and datasets of the experiment, and analyzes the subjective effect and objective index respectively. This paper implements fogging scenes and day to night scene transitions. Compared with subjective effects and objective parameters, the algorithm in this paper achieves good performance compared to other algorithms.

Acknowledgment. This work was supported in part by the National Key Research and Development Program of China (Grant No. 2017YFB1302401), the National Natural Science Foundation of China (Grant No. 61573002), and the Science and Technology Project of State Grid Hubei Power Co., Ltd. (Grant No. 52153318004G).

References

1. Isola, P., Zhu, J.Y., Zhou, T., et al.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1125–1134 (2017)
2. Xiao, J., Luo, L., Liu, E., et al.: Single-image Dehazing algorithm based on convolutional neural networks. In: 2018 24th International Conference on Pattern Recognition, ICPR 2018, August, Beijing, pp. 1259–1264 (2018)
3. Xiao, J., Tian, H., Zhang, Y., et al.: Blind video denoising via texture-aware noise estimation. *Comput. Vis. Image Underst.* **169**, 1–13 (2018)
4. Goodfellow, I., Pouget-Abadie, J., Mirza, M., et al.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, pp. 2672–2680 (2014)
5. Wang, C., Zheng, H., Yu, Z., Zheng, Z., Gu, Z., Zheng, B.: Discriminative region proposal adversarial networks for high-quality image-to-image translation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11205, pp. 796–812. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01246-5_47
6. Mirza, M., Osindero, S.: Conditional generative adversarial nets. arXiv preprint [arXiv:1411.1784](https://arxiv.org/abs/1411.1784) (2014)
7. Zhu, J.Y., Park, T., Isola, P., et al.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2223–2232 (2017)
8. Chen, Q., Koltun, V.: Photographic image synthesis with cascaded refinement networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1511–1520 (2017)
9. Dosovitskiy, A., Brox, T.: Generating images with perceptual similarity metrics based on deep networks. In: Advances in Neural Information Processing Systems, pp. 658–666 (2016)
10. Freedman, D., Kisilev, P.: Object-to-object color transfer: optimal flows and smpsp transformations. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 287–294. IEEE (2010)
11. Tsai, Y.H., Shen, X., Lin, Z., et al.: Sky is not the limit: semantic-aware sky replacement. *ACM Trans. Graph.* **35**(4), 1–11 (2016)
12. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2414–2423 (2016)
13. Chen, D., Yuan, L., Liao, J., et al.: Stylebank: an explicit representation for neural image style transfer. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1897–1906 (2017)
14. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1501–1510 (2017)
15. Scharstein, D., et al.: High-resolution stereo datasets with subpixel-accurate ground truth. In: Jiang, X., Hornegger, J., Koch, R. (eds.) GCPR 2014. LNCS, vol. 8753, pp. 31–42. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11752-2_3
16. Laffont, P.Y., Ren, Z., Tao, X., et al.: Transient attributes for high-level understanding and editing of outdoor scenes. *ACM Trans. Graph. (TOG)* **33**(4), 149 (2014)
17. Choi, L.K., You, J., Bovik, A.C.: Referenceless prediction of perceptual fog density and perceptual image defogging. *IEEE Trans. Image Process.* **24**(11), 3888–3901 (2015)



An Embedded Real-Time Monocular SLAM System Utilizing a Dynamically Reconfigurable Processor

Koki Kawashima^(✉) and Koyo Katsura

System Design Program, Graduate School Engineering, Kogakuin University,
1-24-2 Nishi-Shinjuku Shinjuku-ku, Tokyo 163-8677, Japan
gml18003@ns.kogakuin.ac.jp,
koyokatsura@cc.kogakuin.ac.jp

Abstract. In this paper, we propose an Embedded Real-time Monocular SLAM (Simultaneous Localization and Mapping) System for an autonomous indoor mobile robot. Autonomous mobile robots must be able to estimate and maintain the pose of the robot and the map of the environment at the same time. SLAM performs those tasks using one or more external sensors (e.g., LiDAR, Camera, and Inertial Measurement Unit). The previous SLAM system had problems with a sensor size, high power consumption, and high cost. Thus, it is hard to implement on a small indoor robot. We propose an Embedded (small size, low power consumption, and low cost) Real-time Monocular SLAM System which combines an ORB feature extraction-based SLAM (ORB-SLAM), a monocular camera, and a dynamically reconfigurable processor (DRP). This system realizes real-time (30 fps over) and low-power (less than 2 W) SLAM utilizing the hardware accelerating function of DRP. In the future, we will examine the speed-up of all processing and build it into a device.

Keywords: Visual SLAM · Dynamically reconfigurable processor · FPGA · Monocular camera

1 Introduction

Autonomous mobile robots are expected to solve the social problems of the decrease of workers and increase of aging population. According to a market trend report released by Boston Consulting Group (BCG), the global robotics market will grow at CAGR (a compounded growth rate) of 11.2% until 2025 [1], and the market size will be \$67 billion. In this market report, BCG divides the robot industry into four segments (military, industrial, commercial, and personal) and forecasts each growth. The most growing segment is the personal segment at CAGR of 17.0%. The personal segment's robots are used for entertainment, cleaning, education, security, and household applications. Therefore, the indoor use robots will increase. The future personal robots that help people need to be able to run autonomously and need to be built in small size.

When autonomous mobile robots are running, the robots must be able to estimate and maintain the pose and the map of the environment at the same time. Recently, many researchers are actively researching and developing SLAM (Simultaneous

Localization and Mapping) technology for performing those tasks in real-time. SLAM uses one or more external sensors (e.g., LiDAR, Camera, and Inertial Measurement Unit). However, the conventional systems have a lot of problems; sensor size, high power consumption, and high cost. Thus, it has been hard to implement on a small indoor robot. We propose an Embedded Real-time Monocular SLAM System that realizes real-time (30 fps over), low-power (less than 2 W), and small in size.

This paper is organized as follows: Section 2 presents a method of visual-based SLAM. Section 3 shows the related works. The Embedded Real-time Monocular SLAM System is explained in Sect. 4. Sections 5 and 6 are analysis and evaluation results of the method. This paper concludes and describes future works in Sect. 7.

2 Visual SLAM

Visual SLAM is one of the 3D SLAM algorithms using a camera image. Generally, the Visual SLAM uses a monocular camera, stereo camera, RGB-D camera as external sensors. The advantage of Visual SLAM is it enables to use a camera that is cheaper, smaller, and lighter than the others (e.g., LiDAR). In addition, Visual SLAM can be expected to generate a map including information of objects (e.g., chair, desk, and light) by performing object recognition using deep learning. Thus, Visual SLAM is a better fit for a small indoor robot system. However, the processing speed of the Visual SLAM needs to be improved. If the complicated processing can be speed up, we can use Visual SLAM in embedded devices.

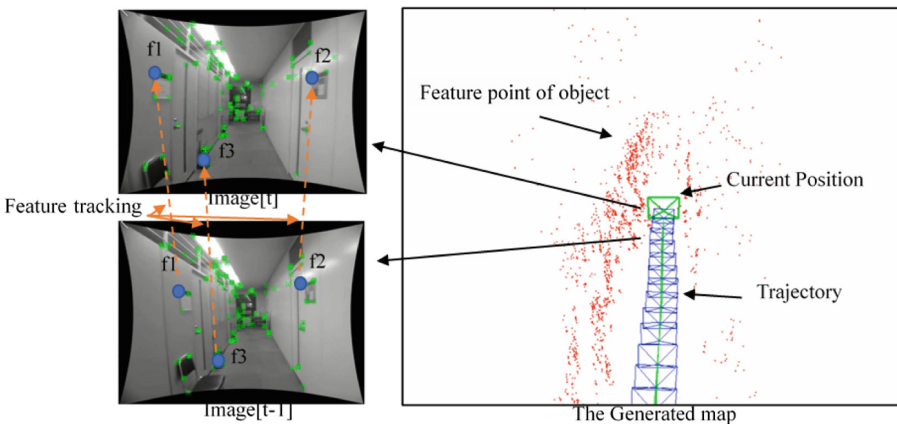


Fig. 1. An example of Visual SLAM (Color figure online)

Figure 1 shows an example of Visual SLAM. The upper left image is an input image at time t , and the lower left image is an input image at time $t - 1$. In both images, the green points are the extracted feature points of objects. The right image is the generated map of the environment. The green line is a trajectory of the estimated

positions, the green square is the current position at time t , and the red points show the positions of objects.

Visual SLAM processing is as follows: (1) Input an image from camera, (2) Extract a feature point on the image, (3) Track the feature point, (4) Estimate the pose, (5) Update the map of the environment, (6) Detect a Loop and Correct the map.

3 Related Works

In this section, we describe related works on real-time visual SLAM. Recently, many researchers are actively researching and developing the real-time visual SLAM using various devices and accelerators.

A GPU-Accelerated Real-time SLAM was proposed by Donald Bourque [2]. This research realizes a speed increase of 33% by using GPU-acceleration with CUDA; it is implemented on Nvidia Jetson TX2. However, the power consumption of the device is 7.5 W, and the price of the device is over \$450.

Another research presents an energy-efficient accelerator for visual-inertial odometry (VIO) that enables autonomous navigation of miniaturized robots [3]. The presented entire VIO system is fully integrated on a chip to reduce the energy consumption and footprint. This system realizes 28–171 fps at 753×480 pixels and the average power consumption of 24 mW. However, since this system is integrated on ASIC, it cannot easily update parameters and processing algorithms. Additionally, it is difficult to decrease the price because of the high production cost.

Some studies have already been conducted to speed up SLAM processing. For example, an FPGA-based ORB feature extraction (the ORB feature extraction is a part of ORB-SLAM) was proposed [4]. This research examines the speed-up of an ORB feature extraction. The proposed method realizes 488 fps at 640×480 pixels.

To build on a small indoor robot, it is necessary to realize small, low-power consumption and low-cost system. Thus, we propose the Embedded Real-time Monocular SLAM System that is utilizing the hardware accelerating technology for the whole of the processing.

4 Embedded Real-Time Monocular SLAM System

We propose an Embedded Real-time Monocular SLAM System. The proposed system combines an ORB feature extraction-based SLAM (ORB-SLAM), a monocular camera, and a dynamically reconfigurable processor (DRP). This system realizes real-time SLAM utilizing a hardware acceleration and a dynamic reconfiguration of the DRP. In this section, we explain the details of ORB-SLAM, DRP and Hardware Acceleration.

4.1 ORB-SLAM

In this research, we develop a SLAM system based on ORB-SLAM2. ORB-SLAM2 is an open-source SLAM system for Monocular, Stereo, and RGB-D camera, and it is one of feature-based visual SLAM [5]. The ORB (Oriented FAST and Rotated BRIEF) is a

faster and efficient feature extraction than others [6]. Moreover, it is built on the FAST keypoint detector and the BRIEF descriptor. The ORB-SLAM can realize the reduction of data volume, speeding-up, and robust by adopting the ORB feature extraction.

Figure 2 shows the flow of ORB-SLAM2 processing. The ORB-SLAM2 is composed of three main threads: tracking, mapping, and loop closing. Each thread is as follows;

- (1) The tracking thread has two tasks; Frame task and Track task. This thread performs the image input, ORB feature extraction, and the pose estimation. After that, a current map and the detected feature points are compared. If it finds a key point above a threshold value, it generates a new keyframe.
- (2) The mapping thread has six tasks; New Keyframe Process task, Map Point Culling task, New Map Points Creation task, Searching in Neighbors task, Local Bundle Adjustment task, and Keyframe Culling task. This thread performs the map generation or the map updating when a new key point is found.
- (3) The Loop closing thread has three tasks; Loop Detection task, Sim3 Computation task, and Loop Correct task. This thread performs the loop detection and the correction.

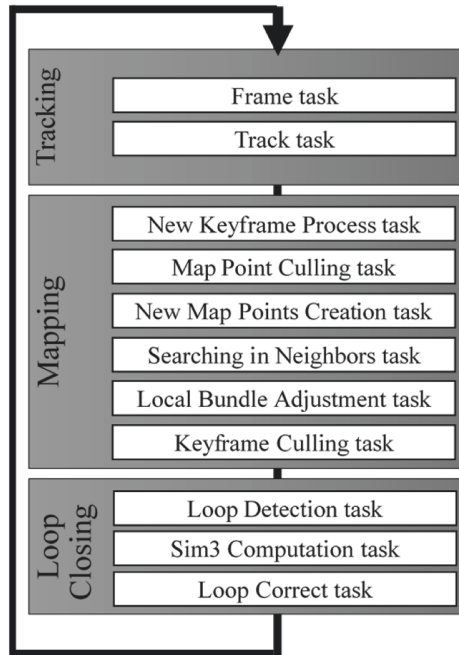


Fig. 2. The flow of ORB-SLAM2 processing

4.2 Dynamically Reconfigurable Processors

In this research, we conduct real-time SLAM processing utilizing a dynamically reconfigurable processor (DRP). A DRP product is produced by Renesas Electronics Corporation; it is called “RZ/A2M”. The DRP is programmable hardware which has both the flexibility of software and the speed of hardware. DRP can speed up some complicated processing by hardware acceleration technology, and DRP can change the hardware configuration at any time and at high speed (it is called “Dynamic Reconfiguration”). Thus, even if the rewritable hardware size is small, the DRP enables to implement the various and large hardware configurations (such as a hardware configuration that include all SLAM processing) by using the dynamic reconfiguration. Additionally, DRP can perform the DRP’s function (e.g., Dynamic reconfiguration) energy-efficiently by providing a direct connected DMA controller. Therefore, DRP enables to realize the embedded real-time monocular SLAM system by using the hardware acceleration and the dynamic reconfiguration of DRP efficiently.

The primitive unit of DRP core is called “Tile”, and a DRP core consists of six Tiles. The Tile has PEs (Processing elements), Mems (Memory elements), and input/output FIFOs. The structure of a Tile is shown in Fig. 3. The PE has an 8 or 16-bit ALUs (Arithmetic and Logic Unit) and a register. The Mem has a two-port data memory and a register [7, 8].

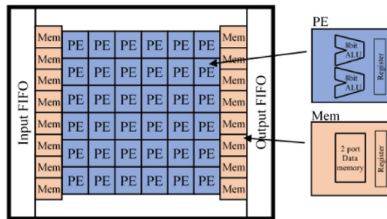


Fig. 3. Structure of a Tile

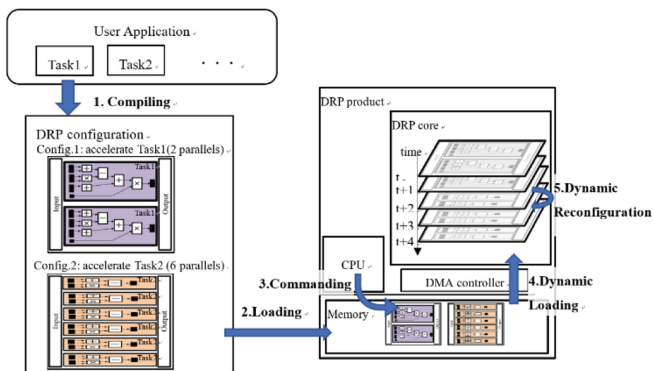


Fig. 4. The flow of DRP functions

The flow of DRP functions is shown in Fig. 4 and are as follows;

1. Compiling user application. Creating the DRP configuration by using HLS tools.
2. Loading a configuration file into the memory of the DRP.
3. Commanding from the CPU and starting of DRP functions when the user application needs the acceleration.
4. Dynamic loading; Loading the configuration data from the memory to DRP core via the DMA controller.
5. Dynamic Reconfiguration; Rewriting the hardware configuration of the DRP core at any time. (e.g., config. 1 to config. 2)

4.3 Hardware Acceleration

Hardware acceleration is realized by using parallel processing and pipeline processing. In this research, we examine to speed-up some complicated processing of the SLAM tasks which is taking a lot of time. Table 1 shows the processing time of each SLAM tasks from the prior art [9]. It was measured on the following experiment environment; CPU: Intel Core i7-870@2.39 GHz, Memory: 12 GB, OS: Ubuntu16.04 (64-bit), and Input image size: 376×1241 pixels. According to the survey, the heaviest task is the Loop Correct task. However, the number of calls of the Loop Correct task is 4; it means the Loop Correct task is performed only when the loop is detected. It does not affect the real-time processing. To realize a real-time SLAM, acceleration of other tasks that are always running need to be examined. The heavy tasks are Local Bundle Adjustment task, Searching in Neighbors task, New Map Points Creation task, and Frame task. We examine the hardware acceleration for those tasks and realize real-time SLAM.

Table 1. Processing time of each SLAM tasks^a

Task name		Number of calls	Processing time (second)
Tracking thread	Frame task	4541	0.029
	Track task	4541	0.012
Mapping thread	New Keyframe Process task	1528	0.017
	Map Point Culling task	1528	0.000
	New Map Points Creation task	1528	0.041
	Searching in Neighbors task	1528	0.047
	Local Bundle Adjustment task	1528	0.097
	Keyframe Culling task	1528	0.004
Closing thread	Loop Detection task	1527	0.008
	Sim3 Computation task	244	0.001
	Loop Correct task	4	1.603

^aCited from [9]

5 Evaluation

The proposed Embedded Real-time Monocular SLAM System is realized utilizing the hardware acceleration and the dynamic reconfiguration of DRP. Thus, our ideal next step is to experiment it using a DRP product. However, the DRP development tools are not ready yet. Thus, in this research, we implemented a prototype on a Xilinx Zynq FPGA (field-programmable gate array); and simulated the function of DRP. FPGA is a programmable processor as well as the DRP. The difference between a DRP and an FPGA is a reconfiguration time and a rewritable hardware size. We measure the difference and estimate the total performance.

5.1 Hardware Design

Figure 5 shows the hardware design of the evaluation device. In this device, we used Digilent ZYBO-Z7-20 as a main board. ZYBO-Z7-20 is an embedded software and digital circuit development board built around the Xilinx zynq-7020 [10]. The Zynq-7020 is based on the Xilinx All Programmable System-on-Chip (AP SoC) architecture, and it has a Processing System (PS) using Dual core ARM processors and a Programmable Logic (PL) using an FPGA block. The number of Look-up Tables (LUTs) of PL is 53,200, the Flip-Flops (FF) of PL is 106,400, and the Block RAM is 630 KB. This board mounts a USB Serial connector, an Ethernet Connector, and a MIPI CSI-2 compatible Pcam connector. This device is connecting a Sony IMX219PQ camera module via MIPI CSI-2 IF as the monocular camera, and it is attached to Entaniya 165 wide-lens to realize a wide viewing angle.

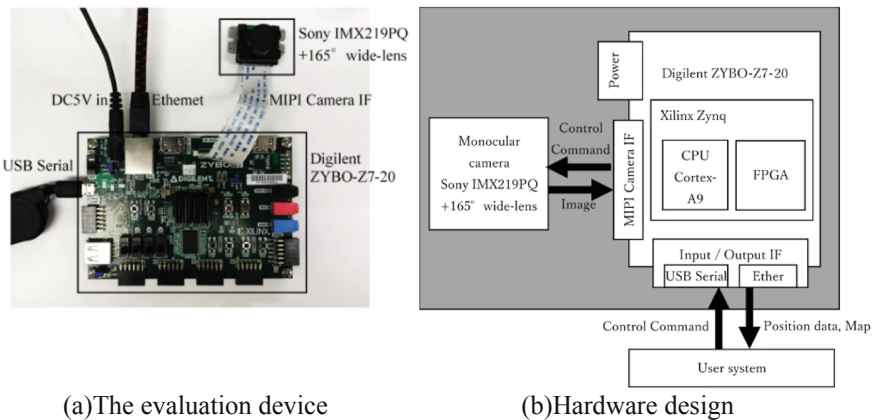


Fig. 5. Hardware design of the evaluation device

5.2 Software Design

Figure 6 shows the software design of the evaluation device. This device uses a Linux Kernel as a main OS. The reason for adopting the Linux kernel is that this device needs

the function of a dynamic reconfiguration for simulating the DRP’s functions. The Linux kernel version 4.10 or later supports the FPGA Region. The FPGA Region is an FPGA management API, and it associates an FPGA Manager and a bridge with a reprogrammable region of an FPGA. The FPGA Region can be reconfigured without powering down. This device simulates a function of the DRP by using FPGA Region.

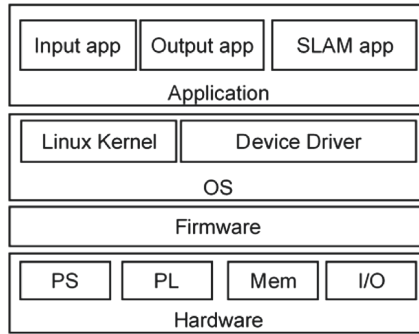


Fig. 6. Software design of the evaluation device

6 Evaluation Results and Discussion

In the current research status, we evaluated the proposed system and examined the speed up of a part of SLAM processing. The evaluation results are as follows.

6.1 The Device Performance

We evaluated the performance of each device (DRP, FPGA and ARM) by performing a Harris corner detection. The device performance is shown in Table 2. The clock frequency of the DRP is 264 MHz, the processing time is 6.5 ms and the power consumption is 1.22 W. The clock frequency of the FPGA is 100 MHz, the processing time is 3.3 ms and the power consumption is 2.30 W. As a result, the DRP can reduce power consumption less than the FPGA. However, the DRP’s processing time is twice times longer than FPGA. The reason is that the difference between programmable hardware had affected.

In addition, we measured the difference of programmable hardware (see Table 3). As a result, the FPGA’s hardware size is 53,200 LUTs and the reconfiguration time is 44 ms. The DRP’s hardware size is as scale as 9,000 LUTs of FPGA and the reconfiguration time is approximately 0.2 ms. Although the DRP has only one-sixth hardware size of the FPGA, it can perform the large processing by using the high-speed reconfiguration.

Table 2. The device performance

	Clock freq. (MHz)	Processing time (ms)	Power consumption ^b (W)
DRP ^a	264	6.5	1.22
FPGA	100	3.3	2.30
ARM Cortex-A9	667	207.0	2.30

^aUse the parallel processing and the dynamic reconfiguration.

^bTotal board power.

Table 3. The deference of a programmable hardware

	Rewritable hardware size (LUTs)	Reconfiguration time (ms)
DRP	As scale as 9,000	About 0.2
FPGA	53,200	44 ^a

^aReference value that is cited from [11]

6.2 Hardware Acceleration

We examined the speed-up of FAST (Features from Accelerated and Segments Test) corner detection, which is a part of SLAM processing. Table 4 and Fig. 7 show the results. The processing time of Hardware Acceleration is 0.94 ms and the throughput is 1064 frame per second. We compared the Hardware Acceleration with ARM Cortex-A9 and Intel Core i7-4650U@1.70 GHz. Compared with ARM, the Hardware Acceleration can speed-up by 12.6x. Compared with Intel, the Hardware Acceleration can speed-up by 1.01x.

Table 4. Hardware acceleration of FAST corner detection

	Processing time (ms)	Throughput (FPS)	Improvement
Hardware Acceleration	0.94	1064	–
ARM Cortex-A9	11.82	85	12.6x
Intel Core i7	0.95	1053	1.01x



Input image (320x240 pixels)



Detected feature point

Fig. 7. Input image and the results

7 Conclusion and Future Work

Visual SLAM is a better fit for an autonomous indoor mobile robot system. However, the processing speed needs to be improved. In this paper, we proposed the Embedded Real-time Monocular SLAM System for an autonomous indoor mobile robot. The proposed system consists of an ORB feature extraction-based SLAM (ORB-SLAM), a monocular camera, and a dynamically reconfigurable processor (DRP). This system realizes real-time SLAM utilizing a hardware acceleration and a dynamic reconfiguration of the DRP. In the current status, we evaluated that the proposed system could realize small and low-power consumption (less than 2 W). In future, we would examine the speed-up of all processing and build into the DRP device.

References

1. Frank Tobe. <https://robohub.org/latest-research-report-shows-10-4-cagr-for-robotics-to-2025/>. Accessed 21 July 2019
2. Bourque, D.: CUDA-accelerated ORB-SLAM for UAVs (2017)
3. Souleiman, A., Zhang, Z., Carlone, L., Karaman, S., Sze, V.: Navion: a fully integrated energy-efficient visual-inertial odometry accelerator for autonomous navigation of nano drones. In: 2018 IEEE Symposium on VLSI Circuits, pp. 133–134 (2018)
4. Weberruss, J., Kleeman, L., Boland, D., Drummond, T.: FPGA acceleration of multilevel ORB feature extraction for computer vision. In: IEEE 2017 27th International Conference on Field Programmable Logic and Applications (FPL), pp. 1–8 (2017)
5. Mur-Artal, R., Tardós, J.D.: ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE Trans. Robot.* **33**(5), 1255–1262 (2017)
6. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.R.: ORB: an efficient alternative to SIFT or SURF. In: *ICCV*, vol. 11, no. 1, p. 2 (2011)
7. Suzuki, M., et al.: Stream applications on the dynamically reconfigurable processor. In: Proceedings of the IEEE International Conference on Field-Programmable Technology (IEEE Cat. No. 04EX921), pp. 137–144 (2004)
8. Renesas Electronics Corporation Homepage. <https://www.renesas.com/jp/ja/products/programmable/accelerator-type-drp.html>. Accessed 29 July 2019
9. Ushiomura, K., Ohkawa, T., Ootsu, K., Baba, T., Yokota, T.: Processing time analysis for accelerating Visual SLAM software. In: Proceedings of the 80th National Convention of IPSJ 2018.1, pp. 125–126 (2018)
10. Digilent Homepage. <https://reference.digilentinc.com/reference/programmable-logic/zybo-z7/reference-manual>. Accessed 21 July 2019
11. Kohn, C.: Partial reconfiguration of a hardware accelerator on Zynq-7000 all programmable SoC devices. Xilinx, XAPP1159 (v1. 0) (2013)



Writer Identification Based on Combination of Bag of Words Model and Multiple Classifiers

Ayixiamu Litifu^{1,2}, Yuchen Yan¹, Jinsheng Xiao^{1(✉)}, Hao Jiang¹,
Weiqing Yao³, and Jihua Wang⁴

- ¹ School of Electronic Information, Wuhan University, Wuhan 430072, China
{Ayixia, yyc, xiaojs, jh}@whu.edu.cn
- ² School of Physics and Electronic Engineering, Xinjiang Normal University,
Urumqi 830054, China
- ³ State Grid Hubei Information & Telecommunication Company Limited,
Wuhan 430077, China
ywq1005@whu.edu.cn
- ⁴ Beijing Huanyu Hongye S&T Development Co., Ltd., Beijing 100000, China

Abstract. In this paper, an efficient approach for text-independent writer identification using bag of words model and the combination of multiple classifiers is proposed. First of all, a bag of words model is established by extracting sub-images from the original handwriting image. Then, features are extracted by moment method, direction index histogram method and simplified Wigner method respectively to calculate the distance between the sub images having the same labels. Finally, the handwriting classification task is completed by means of feature fusion and multi-classifier combination. To evaluate this approach, writer identification is conducted on IAM English database. Experimental results revealed that the proposed writer identification algorithm with small number of characters and unconstrained contents achieves interesting results as compared to those reported by the existing writer recognition systems.

Keywords: Writer identification · Bag of words · Text independent · Multiple classifiers combination

1 Introduction

Writer identification refers to a document identification technology that identifies the writer's identity through handwritten text information. Handwriting reflects the special writing behavior of writers for a long time and handwriting identification technology has played an important role in historical document analysis, identification of judicial suspects and classification of ancient manuscripts for several decades.

At present, writer identification can be divided into two categories: online handwriting identification and offline handwriting identification. The former one relies on real-time information acquired by specific terminal equipment for identification, such as people's writing speed, acceleration, pressures and stroke order. The latter one is based on the text information written on the paper and uses the information such as

handwriting shape, angle and texture to confirm the writer's identity. Compared with online writer identification, offline handwriting provides less information and is more difficult to identify. This paper studies the offline handwriting writer identification method.

There are two kinds of features that can be extracted in writer identification: local features and global features. Scale Invariant Feature Transform (SIFT) or SIFT-like descriptors [1, 2], Local binary pattern (LBP), Local Ternary Patterns (LTP) and Local Phase quantization (LPQ) [3] are the most common local feature extraction methods. Global features are extracted from an input image at the document level and paragraph level. Moreover, there are several studies on combining local and global features.

The proposed approach deals with offline writer identification using the local features. In fact, the number of sample characters required in most handwriting related tasks is relatively small. In this case, increasing the number of samples will lead to a decline in system performance. The experimental results show that the proposed method can avoid the negative impact brought by increased number of writers.

The main contributions of this paper are reflected in three aspects:

- (1) Extracting local features at the sub-region level to generate global features.
- (2) Studying the text independent features based on text dependent bag of words model.
- (3) Combination strategy of multiple classifiers.

The remaining part of this paper is organized as follows: Sect. 2 reviews the related work in the previously published articles. Section 3 describes in detail the flow of text-independent writer identification algorithm based on bag of words model and multiple classifiers. The performances and evaluations are given in Sect. 4. Conclusions and outlook of future work are presented in Sect. 5.

2 Related Work

The growth of artificial intelligence and pattern recognition has greatly promoted the development of writer identification technology [4]. Paper [5, 6] presents a survey of the literature on writer identification schemes and techniques up till 2016, and summarizes the current situation of offline text-independent writer identification methods. This section briefly reviews related works about extracting local features and constructing codebooks.

As off-line writer identification requires writer-specific features, the typical bag-of-words model with the SIFT feature have been used in ref. [7–9]. Recently, paper [10] presents a texture based approach which divides a given handwriting into small fragments and considers it as a texture. In order to describe the local features well, some works attribute the extracted local features to various size of codebooks [11–13].

Recently, innovative approaches are being developed and published. In 2017, ref. [14] proposed a robust offline writer-identification system using bagged discrete cosine transform descriptors; ref. [15] proposed two curvature-free features: run-lengths of local binary pattern and cloud of line distribution features for writer identification. In 2018, ref. [16] proposed an end-to-end deep-learning method and ref. [17] proposed

writer identification method based on handwritten stroke analysis. The above-mentioned approaches are more suitable for writer identification tasks with fewer samples or many characters written on the sample. In the word bag generation phase in this paper, the segmentation work of high-frequency sub-images is not affected by the window size and shape transformation, and the number of sub-images to be extracted is far less than that in the above papers. Obviously, the proposed algorithm has certain advantages in sub-image segmentation, feature extraction and classification parts.

3 Proposed Approach

The proposed algorithm is implemented by two steps: preprocessing and testing. The preprocessing part mainly includes binarization of the original image, sub-image extraction, labeling and generation of bag of words. The testing part mainly completes the operations of feature extraction, feature fusion and multi-classifier combination. The implementation flow is shown in Fig. 1 below.

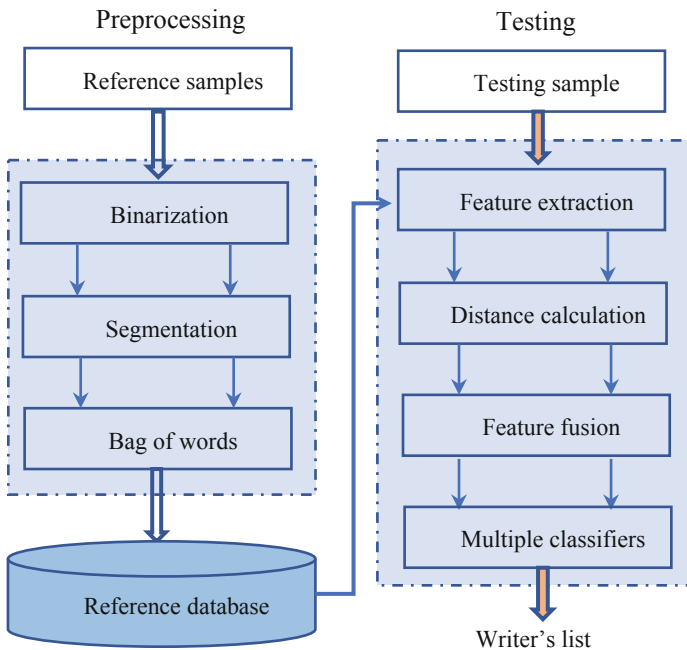


Fig. 1. Writer identification flowchart

In this system, all scanned handwriting images are divided into two groups: reference samples and test samples. Then, all images are converted into binary images, and removed various noises, rows and grids on it. Then, high-frequency sub-images are extracted and labeled with symbols after normalization. All marked sub-images form a

writer specific “bag of words” model. In this part, the segmentation of sub-images is the basis of the word bag model, and labeling is for easy retrieval. In the testing part the system first retrieves sub-images with the same mark between the reference sample and the testing sample. Then, for the sub-images matched by the labels, the moment method, direction index histogram method and Wigner method are used to extract features and calculate the feature distance respectively. Finally, the input image closest to the test sample is determined by feature fusion and multi-classifier combination approach.

3.1 Word Segmentation

The original image is converted into a binary image by Otsu transformation, and the weak texture regions are selected by the thresholding function that is deduced based on the normal distribution [18]. The sub-images are segmented by using a rectangular window of any size. The previous experimental results showed that compared with sub-regions at paragraph level and letter level, sub-images at word level can better reflect personal writing style [12]. Therefore, the size of the sub-images are controlled at 1–6 bytes and normalized to a matrix with a size of 64*64. The word bag of reference sample always includes redundant sub-images to increase the matching probability with existing patterns in testing sample. When it comes to extracting sub-images, the grammatical structure and writing characteristics of certain language should be studied. Considering the small number of handwritten characters and the unconstrained contents on samples, the bag of words contains prefixes, suffixes, syllables, simple words and letters of sub-images. Inspired by the high-frequency writing mode proposed in paper [11–13], this phase proposed a sub-image strategy with semantic information. After segmentation, labeling and normalization process, all sub-images will generate a word bag. The four steps are shown in Fig. 2 below.

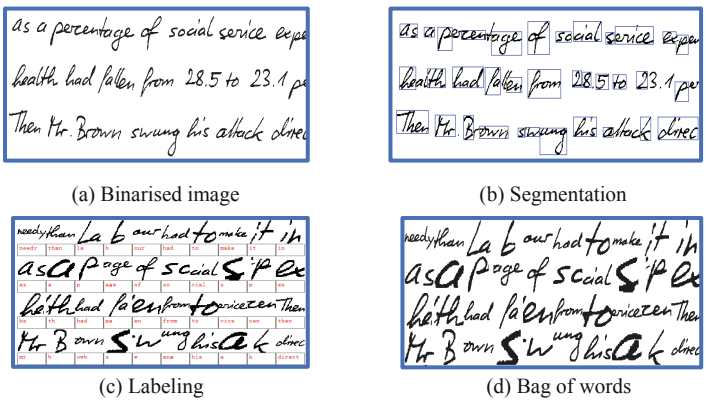


Fig. 2. Bag of words models of different writers

3.2 Feature Extraction

Three types of feature extraction algorithms used in this paper: moment method, direction index histogram method and Wigner method.

3.2.1 Moment Method

Moment feature is a basic method used to represent the shape of objects and to identify invariant objects in the fields of computer vision and pattern recognition [19]. In writer identification, the contour, gradient and deviation of characters are important features that reflect writing style. Various forms of sub-images are shown in Fig. 3 below

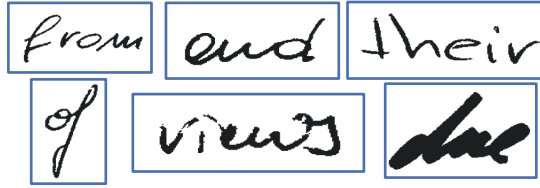


Fig. 3. Sub-images with different physical characteristics

Geometric moments are widely used features in this field and the geometric features are invariant under translation, scaling and stroke width. They are explicitly corresponding to human perception of shape and distributing their values in small dynamic ranges. For a digital image $f(x, y)$ with a size of $M \times N$, the formula for calculating the $p + q$ order geometric moment can be written as follows:

$$m_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} x^p y^q f(x, y) \quad (1)$$

The center of gravity (X, Y) of an object can be obtained from the zero-order and first-order geometric moments:

$$\begin{cases} X = \frac{m_{10}}{m_{00}} \\ Y = \frac{m_{01}}{m_{00}} \end{cases} \quad (2)$$

The center moment U_{pq} can be obtained by taking the center of gravity as the origin of coordinates:

$$U_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x - X)^p (y - Y)^q f(x, y) \quad (3)$$

There are $U_{00} = m_{00}$ and $U_{01} = U_{10} = 0$ for zero order moments and first-order moments. The central moment is constant relative to the displacement and the low-order moment has obvious physical meaning. The zero-order moment represents the sum of the gray values of all pixels in the image or the number of black pixels for binary images. The secondary moment refers to the variance, where U_{20} and U_{02}

represent the extension of black spots in the image in the horizontal and vertical directions respectively, and the secondary moment U_{11} represents the inclination of the object. The third-order moments U_{30} and U_{03} represent the deflection degree of the object in the horizontal and vertical directions. However, the third-order moments U_{21} and U_{12} represent the extension equilibrium degree of the object in the horizontal and vertical directions. According to the above features, the overall shape of the sub-images can be described. More importantly, the value range of other features can be mapping in $[0, 1]$ except the word position direction feature is in the range of $[-1, 1]$.

3.2.2 Direction Index Histogram Method

This is a template matching method to extract sub-image grid features considering the shape of the input image [5]. This method firstly divides the input image into 8×8 grids evenly, then divides each grid into 8×8 sub-regions to calculate contour points in four directions, and obtains 8×8 four-dimensional histograms n_{ijk} of the input image, where $i, j = 1, 2, \dots, 8$ represent grid positions, $k = 0, 1, 2, 3$ represent directions, and the obtained histograms reflect contour shapes in the sub-regions. The method for determining the direction of local strokes in this algorithm is as follows: When one (three) of the four adjacent points of the contour point is zero, take the vertical direction of the neighborhood point relative to the current contour point as the stroke direction. When the contour points have two four-neighborhood points of zero, if the two neighborhoods are connected, their connection direction is taken as the stroke direction. Otherwise, the vertical direction of their connection is taken as the stroke direction. The case where all four neighborhood points are equal to zero will not be considered. Then, the Gaussian function with mean square deviation of $\sigma^2 = 40$ is used to perform spatial smoothing to n_{ijk} on an 8×8 grid plane. At the same time, the values of 4×4 points are sampled as features, and the chain code generates $4 \times 4 \times 4 = 64$ bit feature vectors f_{uvk} :

$$f_{uvk} = \sum_i \sum_j n_{ijk} \exp \left[\frac{-(x_i - x_u)^2 - (y_j - y_v)^2}{2\sigma^2} \right] \quad (4)$$

Where $u, v = 0, 1, 2, 3$ and (x_u, y_v) represents the coordinates of the sampling point in the character image and (x_i, y_j) is the coordinate of the 8×8 grid center point. After obtaining the 64-bit feature vector f , the distance $d(f_1, f_2)$ between two feature vectors can be calculated:

$$d_1(f_1, f_2) = \frac{\sum_{i=1}^{64} |f_{1i} - f_{2i}|}{\sqrt{\sum_{i=1}^{64} f_{1i} \sum_{j=1}^{64} f_{2j}}} \quad (5)$$

3.2.3 Simplified Wigner Distribution Method

Wigner distribution of images is a joint representation of Spatial/Spatial Frequencies which suitable for representing texture features of images. However, the amount of storage and computation required to calculate Wigner distribution is very large in general, and simplification measures must be taken. Calculating a two-dimensional

Wigner distribution function $W_f(x, y, u, v)$ for the 64×64 character image $f(x, y)$ as handwriting features:

$$\begin{cases} W_f(x, y, u, v) = \iint R_f(x, y, \alpha, \beta) e^{-2\pi j(\alpha u + \beta v)} d\alpha d\beta \\ R_f(x, y, \alpha, \beta) = f\left(x + \frac{\alpha}{2}, x + \frac{\beta}{2}\right) f^*\left(x - \frac{\alpha}{2}, x - \frac{\beta}{2}\right) \end{cases} \quad (6)$$

Where $R_f(x, y, \alpha, \beta)$ represents a local correlation function with the (x, y) as the center and the (α, β) as the displacement. The strokes of the characters are roughly distributed in horizontal, vertical and two diagonal directions. Then, in order to reduce the resolution and offset the influence of stroke thickness, the local correlation coefficient is smoothed and normalized in the spatial domain. Actually, the smoothing process is to project 64×64 points of data to $4 \times 4 = 16$ spatially positioned weighted windows. Finally, FFT is used to calculate the power spectrum for the correlation coefficients in four direction, and the total feature dimension of a character is $16 \times 4 \times 4 = 256$. This is the simplified Wigner distribution. The distance between the two 256-dimensional feature vectors x_1 and x_2 is calculated by the following formula:

$$d_2(x_1, x_2) = \frac{\sum_{i=1}^{256} |x_{1i} - x_{2i}|}{\sqrt{\sum_{i=1}^{256} x_{1i} \sum_{i=1}^{256} x_{2i}}} \quad (7)$$

3.3 Feature Fusion and Multi-classifier Combination

Feature fusion refers to extracting features from a single sub-image in a word bag, and then mapping local features to a common space to obtain global feature vectors of the whole handwriting. The combination of multiple classifiers can be roughly divided into three types: series, parallel and series-parallel hybrid combination. In this paper, series-parallel hybrid combination is adopted. After the reference sample and the test sample are replaced by the word bag model, the system extracts the three features of all sub-images in the word bag and generates feature vectors. In the first step, the most of the samples with small similarity are rejected by the moment based classifier. Then, suspected samples are further classified by a parallel combination of directional index histogram method and simplified Wigner distribution method.

Experiments show that the direction index histogram method is an identification method with high accuracy and fast calculation speed. The Wigner distribution is not easy to exclude similar patterns in order to reduce the verification error rate. Therefore, the reasonable combination of the three algorithms can improve the recognition rate of system.

4 Experimental Results

In the following sections, we will describe the data set and evaluation index used in this paper, then analyzed the influencing factors and proposed the adjusting strategy of parameters. The experimental results and the comparison with previous studies are discussed at last.

4.1 Data Set and Evaluation Metrics

The experiments will be carried out on English benchmark datasets IAM [20], which are publicly available and have been applied in many recently published papers. The database contains handwritten samples from 657 writers. All of the images are divided roughly in half and one of them is used for referencing while the other is used for testing.

The evaluation criteria widely used in image and information retrieval tasks include mean average accuracy (mAP), Soft top-K (TOP-k) and hard top-k methods [6]. Furthermore, these testing methods have several typical comparison strategies such as leave-one-out comparison, 2-fold metric [12] and dissimilarity calculation [3, 10], etc. In this paper, we will use leave-one-out comparison strategy and Top-k evaluation criteria.

4.2 Analysis of Influencing Factors

In the proposed algorithm, the factors that affect the test results include the number of sub-images and writers. Experiments show that the number of samples does not cause a dramatic change in the results. In this paper, a number of 150 handwritings are used to extract sub-images to observe the influence of the number of sub-images. Details are shown in Fig. 4 below.

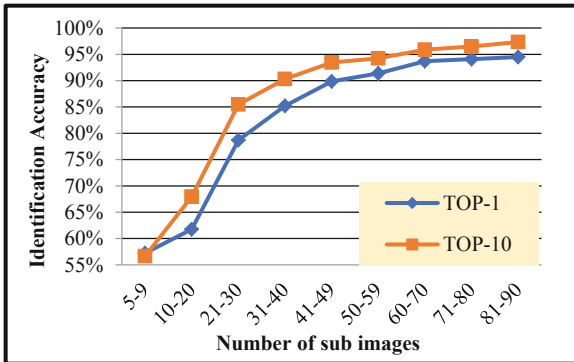


Fig. 4. Influence of number of sub images

As can be seen from Fig. 4, when the number of sub-images is gradually increases from 5 to 50, the identification accuracy increases from 20% to 90%, and keeps a relatively stable value. Although the proposed algorithm does not require writers to write a large page or many characters, the number of sub-images extracted from a single sample required to reach 40 as far as possible. However, compared with papers [12, 13], the proposed algorithm requires fewer sub-images. The main reason is that our sub-images are extracted in units of letters, syllables and words and carry more information about writing styles than character fragments.

4.3 Experiments and Comparison

In this part, we first extract 40 sub-images from each sample on IAM dataset to generate a word bag model. Then gradually increase the number of sub-images and retest failed handwriting samples. In order to improve the stability and robustness of the system, a complete extraction method is adopted for the repeated words and word blocks on a sample. The identification results of the three classifiers and the combined identification results of multiple classifiers are shown in Table 1 below.

As can be seen, the discrimination ability of the combination three classifiers far exceeds the capability of a single classifier. Different classifiers have complementary information for classification patterns. The fuzzy integral method is used to combine the classifier results, and the list of writer is sorted by the highest sequence number method. In order to compare the identification performance on IAM dataset with the results of other papers, Table 2 is listed.

Table 1. Performance comparison on IAM database (%) (650 person)

Classifiers	Top-1	Top-10
Moment Features	82.5	87.4
Direction Index Histogram	86.8	92.1
Wigner Distribution	85.2	90.5
Multi-classifier combination	94.6	98.6

Table 2 showed that the identification results of Top-1 and Top-10 in IAM data set of this method are only inferior to the reference [1], and the overall performance is relatively high.

Table 2. Performance comparison of different approaches on IAM (%) (650 person)

	Top-1	Top-10
Siddiqi [11]	91.0	97.0
Ghiasi [12]	93.7	97.7
Khalifa [13]	92.0	–
Hannad [10]	89.54	96.77
Wu [1]	98.5	99.5
Khan [14]	92.3	–
He [15]	89.9	96.9
Proposed approach	94.6	98.6

5 Conclusion

This paper proposes a novel writer identification approach based on bag of words and multi-classifier combination models. In the preprocessing part, we extracted sub-images based on words, letters and syllables, and established a word bag model. At the

classification decision-making level, three methods such as moment feature, direction index histogram and Wigner distribution are used to extract local features, and the combination model of the aforementioned three classifiers is used to implement writer identification task. In this paper, IAM data set is used to evaluate the algorithm, and the experimental results verified the feasibility and robustness of the proposed method. Experimental results demonstrate that the proposed algorithm with low computational complexity not only has better estimation results, but also outperforms the state-of-the-art methods in most cases.

Acknowledgment. This work is supported by University Scientific Research Program Natural Science Youth Project of Xinjiang Uyghur Autonomous Region (Grant No. XJUDU2019Y032), and the Tender Subject for Key Laboratory Project of Xinjiang Normal University (Grant No. XJNUSYS092018A02).

References

1. Wu, X., Tang, Y., Bu, W.: Offline text-independent writer identification based on scale invariant feature transformation. *IEEE Trans. Inf. Forensics Secur.* **9**(3), 526–536 (2014)
2. Christlein, V., Gropp, M., Fiel, S., Maier, A.: Unsupervised feature learning for writer identification and writer retrieval. In: *Proceedings of the International Conference on Document Analysis and Recognition*, pp. 991–997 (2018)
3. Chahi, A., El Merabet, Y., Ruichek, Y., Touahni, R.: An effective and conceptually simple feature representation for off-line text-independent writer identification. *Expert Syst. Appl.* **123**, 357–376 (2019)
4. Xiao, J., Zou, W., Chen, Y., Wang, W., Lei, J.: Single image rain removal based on depth of field and sparse coding. *Pattern Recogn. Lett.* **116**, 212–217 (2018)
5. Litifu, A., Yan, Y., Xiao, J., Jiang, H., Yao, W.: Text-independent writer identification based on hybrid codebook and factor analysis [J/OL]. *Acta Autom. Sinica* 1–11 (2019). <https://doi.org/10.16383/j.aas.c190121>
6. Tan, G.J., Sulong, G., Rahim, M.S.M.: Writer identification: a comparative study across three world major languages. *Forensic Sci. Int.* **279**, 41–52 (2017)
7. Fiel, S., Sablatnig, R.: Writer retrieval and writer identification using local features. In: *International Workshop on Document Analysis Systems*, pp. 145–149 (2012)
8. Xiong, Y., Wen, Y., Wang, S., Lu, Y.: Text-independent writer identification using SIFT descriptor and contour-directional feature. In: *International Conference on Document Analysis and Recognition*, pp. 91–95 (2015)
9. Khan, F.A., Khelifi, F., Tahir, M.A., Bouridane, A.: Dissimilarity Gaussian mixture models for efficient offline handwritten text-independent identification using SIFT and RootSIFT descriptors. *IEEE Trans. Inf. Forensics Secur.* **14**(2), 289–303 (2019)
10. Hannad, Y., Siddiqi, I., El Youssfi, M., Kettani, E.: Writer identification using texture descriptors of handwritten fragments. *Expert Syst. Appl.* **47**, 14–22 (2016)
11. Siddiqi, I., Vincent, N.: Text-independent writer recognition using redundant writing patterns with contour-based orientation and curvature features. *Pattern Recogn.* **43**(11), 3853–3865 (2010)
12. Ghiasi, G., Safabakhsh, R.: Offline text-independent writer identification using codebook and efficient code extraction methods. *Image Vis. Comput.* **31**, 379–391 (2013)

13. Khalifa, E., Al-Maadeed, S., Tahir, M.A., Bouridane, A., Jamshed, A.: Off-line writer identification using an ensemble of grapheme codebook features. *Pattern Recogn. Lett.* **59** (1), 18–25 (2015)
14. Khan, F.A., Tahir, M.A., Khelifi, F., Bouridane, A., Almotaryi, R.: Robust off-line text independent writer identification using bagged discrete cosine transform features. *Expert Syst. Appl.* **71**, 404–415 (2017)
15. He, S., Schomaker, L.: Writer identification using curvature-free features. *Pattern Recogn.* **63**, 451–464 (2017)
16. Nguyen, H.T., Nguyen, C.T., Ino, T., Indurkha, B., Nakagawa, M.: Text-independent writer identification using convolutional neural network. *Pattern Recogn. Lett.* **121**, 104–112 (2019)
17. Aubin, V., Mora, M., Santos-Peñas, M.: Off-line writer verification based on simple graphemes. *Pattern Recogn.* **79**, 414–426 (2018)
18. Xiao, J., Tian, H., Zhang, Y., Zhou, Y., Lei, J.: Blind video denoising via texture aware noise estimation. *Comput. Vis. Image Underst.* **169**, 1–13 (2018)
19. Mirzapour, F., Ghassemian, H.: Moment-based feature extraction from high spatial resolution hyperspectral images. *Int. J. Remote Sens.* **37**(6), 1349–1361 (2016)
20. Marti, U., Bunke, H.: The IAM-database: an English sentence database for off-line handwriting recognition. *Int. J. Doc. Anal. Recogn.* **5**, 39–46 (2002)

Advances and Applications on Generative Deep Learning Models



Vehicle-Related Scene Understanding Using Deep Learning

Xiaoxu Liu, Minh Neuyen, and Wei Qi Yan ^(✉)

Auckland University of Technology, Auckland 1010, New Zealand
weiqi.yan@aut.ac.nz

Abstract. Automated driving is an inevitable trend in future transportation, it is also one of the eminent achievements in the matter of artificial intelligence. Deep learning produces a significant contribution to the progression of automatic driving. In this paper, our goal is to primarily deal with the issue of vehicle-related scene understanding using deep learning. To the best of our knowledge, this is the first time that we utilize our traffic environment as an object for scene understanding based on deep learning. Moreover, automatic scene segmentation and object detection are joined for traffic scene understanding. The techniques based on deep learning dramatically decrease human manipulations. Furthermore, the datasets in this paper consist of a large amount of our collected traffic images. Meanwhile, the performance of our algorithms is verified by the experiential results.

Keywords: Traffic scene understanding · Deep learning · Automatic driving · Image segmentation · Object detection

1 Introduction

With the development of artificial intelligence, autonomous vehicles have already been associated with the field of computer vision. Due to complex traffic environment, the capability of traffic scene understanding has become a significant indicator of autonomous vehicles.

Scene understanding is a process of cognizing and inferring the environment based on spatial perception [1]. In vehicle-related scene understanding, a scene is the environment in which the vehicle is currently located including location, person, focus, event, and relationships between them. Scene understanding mainly includes object detection and recognition, semantic segmentation, topological relationship exploration and discovery between objects.

Scene information in a video is extremely dense which has great discrepancy and complexity. Recently, owing to the development of deep learning, the use of scene understanding can significantly ameliorate the performance of video analysis, which is a method of using machine perception [2].

In addition, deep learning has an active merit that a myriad of pretrained networks and public datasets have provided benefits for training numerous traffic scenes [19–21]. For vehicle-related scenes, in order to understand the objects, scenes, and events in a video, deep neural networks emulate high-level abstraction from the visual data and

encode them using a robust representation [3]. Therefore, deep learning methods have unique advantages in the field of vehicle-related scene understanding.

For an example, the end-to-end nature of deep learning is one of such advantages, which achieves faster and universal information processing than traditional methods under the premise of a particularly accurate recognition of specific scenes. For autonomous vehicle that demands to understand the information in complex traffic scene, the method of deep learning can effectively satisfy the accuracy and real-time requirements [4].

Automated driving using deep learning is not mature yet to understand visual objects in complex traffic scenes due to the diversities of global traffic rules and transportation facilities [22]. Currently, it is difficult to apply all traffic scenes using only a single algorithm [5]. The two essential branches of scene understanding give us inspirations, object recognition identifies all objects of a predefined category on the image and positions through a bounding box. Semantic segmentation operates at a fine scale, its purpose is to segment images and associate each region with class labels [6]. Albeit these are two similar tasks, few studies currently merged the two categories of work together.

In this paper, automatic image segmentation and vehicle detection for vehicle-related scene understanding are developed using deep learning so as to reduce human workload. The datasets in this paper provide a great deal of traffic scenes. Simultaneously, adjustments and ameliorations have been implemented with the proposed neural network for scene understanding.

In this paper, literature review will be provided in Sect. 2, our methodology is shown in Sect. 3, the experimental results will be demonstrated in Sect. 4, our conclusion will be drawn in Sect. 5.

2 Literature Review

In this paper, we explain the reasons why the characteristics of deep learning play an essential role in scene understanding and why high-quality scene understanding models are often achieved through deep learning. First, the layer-by-layer processing of deep learning enables the model to better express the information in the current traffic scene. By simulating the structure of human brain and its gradual cognitive process, deep learning models obtain higher-level expressions through a linear or nonlinear combination. Therefore, deep learning enables the models to analyze complex traffic scenes, it has a hierarchical structure for information proceeding similar to our human brain, progressively extracts internal features and sufficient model complexity. These characteristics of deep learning enable the proposed model to understand the high-level semantics of traffic scenes (traffic event analysis, logical relationships of objects in traffic scenes). Currently, deep learning can accurately segment lanes to understand road conditions in the scene [7, 19–23]. It is also possible to predict overtaking, lane changing, and braking events by dynamically detecting the positional relationship between the two vehicles [8].

Secondly, the end-to-end characteristic of deep learning has also made an extraordinary contribution to the development of scene understanding. As the number of vehicles globally increases, the complexity of traffic scenes continues rising. Therefore, autonomous vehicles have higher demands in terms of real-time performance. The model

of traffic scene understanding based on deep learning utilizes the end-to-end process to optimize all tasks (vehicle detection, pedestrian detection, path planning, etc.). For example, an end-to-end vehicle controller can detect obstacles in the scene and navigate them accurately following the curved lanes [9].

Thirdly, deep learning algorithms have strong versatility [10]. Faced with the tasks involved in scene understanding, deep learning models do not require redesigning new algorithms for each task like traditional algorithms. Currently, each deep learning algorithm is suitable for a variety of scene understanding. For example, Faster R-CNN model achieves excellent results such as vehicle detection, pedestrian detection, and lane detection.

Finally, deep learning models have active mobility. A mature scheme of autonomous vehicle must contain a large number of functionalities related to scene understanding which utilizes human experience to train a scene so as to understand the scene from scratch. Deep learning models learn neural network parameters from one task and can transfer them very well to another. For example, the deep learning parameters and knowledge learned based on ImageNet dataset can achieve superior results in scene understanding by using other datasets [11].

In this paper, the understanding of vehicle-related scenes is implemented in conjunction with a deep learning-based vehicle detector and a semantic segmenter. Compared with other machine learning methods, deep neural networks can improve the accuracy by increasing the amount of training data and introducing sophisticated methods to betterment efficacy and accuracy [12]. Additionally, deep learning naturally is an end-to-end model, because the visual data is imported directly to the input layer, the well-trained network can export excellent outcomes. Finally, the underlying concepts and techniques using deep learning are universally transferable. Hence, deep learning can be much adaptive to various datasets for vehicle-related scene understanding.

3 Methodology

In order to deeply discover the merits of deep learning in vehicle-related scene understanding, we detail a computable method in this section based on deep learning for semantic segmentation and vehicle detection.

3.1 Vehicle-Related Scene Understanding

For the understanding of vehicle-related scene, a high-performance model is not only to detect and identify single isolated object, but also to understand advanced semantics in the vehicle-related scenes. Therefore, we make full use of deep learning to simulate the characteristics of our human brain. The high-level semantics in complex traffic scenes are employed through layer-by-layer processing and a stepwise abstraction of the feature map.

If we use a hierarchical system to emulate human cognition of the scenes, the entire human cognitive process of information needs to be carried out through several levels. At a lower level, our humans extract visual and auditory information as basic features, which is similar to the idea of extracting concepts from the scene and forming the basic

layer of the ontology; at a higher level, our human brain unifies these features and makes judgement from these features. At the highest level, our human can obtain the implicit semantics of the information through reasoning, which confirms to semantic expression and semantic understanding [13].

Therefore, in vehicle-related scene understanding, the positional relationship between objects in a scene is very useful for understanding the high-level semantics. It is also one of the necessary steps to use deep learning to analyze traffic scenes. The end-to-end nature of deep learning allows the model to handle multiple tasks. Deep learning models can complete multiple progressive tasks, the results of the previous task are used as an aid to the later. In order to explore the vehicle-related scene more deeply, we explore positional relationship of the objects in the scene for semantic segmentation.

Figure 1 utilizes topological relationships to correlate the classes in scene understanding. According to prior knowledge, visual objects such as trees and buildings should normally appear on both sides of the road. Bus lanes are usually drawn on the road. Most vehicles only travel on the road and do not appear on trees or in the sky. The sky is always above all objects, this is the fact that it never changes.

The relationship in the topological map plays a decisive role in scene understanding based on deep learning. According to the topological relationship between objects in the scene, the model can be used to clarify the object positions in the traffic scene. Deep neural network, as a typical ANN model, can achieve more human-like cognition by learning the logical relationship between objects. Moreover, the topological map constrains the range of the output, reduces the output of unrealistic scene, thereby improves the accuracy of scene understanding.

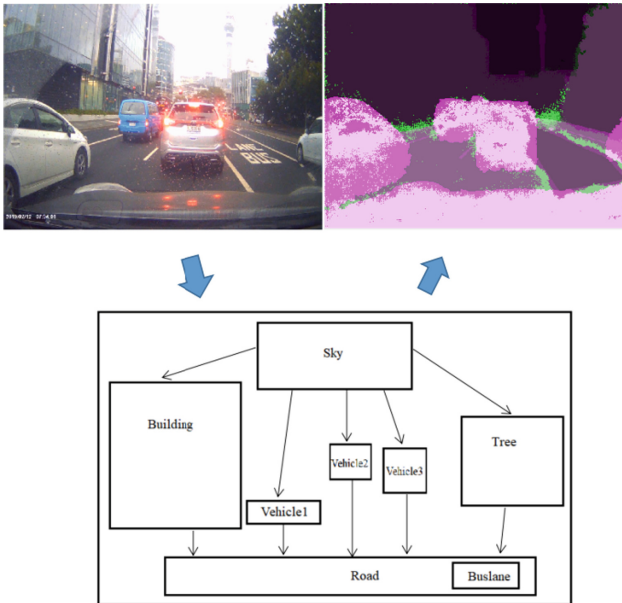


Fig. 1. The topological relationships in the scene

3.2 Methodology of Vehicle-Related Scene Understanding Model

Our model learns parameters by minimizing the value of the loss function. In order to optimize the extremely complex nonconvex function of convolutional neural networks, deep learning models typically exploit feedforward operations to abstract scene information, utilize stochastic gradient descent (SGD), error backpropagation, and chain rules to update the parameters of the model [14].

In the process of feedforward propagation, assume that input x , output y , and the cost $J(\theta)$ are given, the gradient during backward propagation is $\nabla_{\theta}J(\theta)$ [15]

$$J(\theta) = -E\left[\|y - f(x, \theta)\|^2\right] \tag{1}$$

According to the chain rules, if $h = g(k), z = f(g(k)) = f(h)$ then,

$$\frac{dz}{dk} = \frac{dz}{dh} \frac{dh}{dx} = \dot{z} \dot{h} \tag{2}$$

The loss function of SGD is $J(\theta) = L(f_{\theta}(k_i), h_i), (k_i, h_i)$ are samples $i = 1, \dots, m$ with regard to θ in

$$\frac{\partial J(\theta)}{\partial \theta} = 0 \tag{3}$$

If α is a learn rate, we can construct the weight decay function as

$$\theta_{t+1} = \theta_t - \alpha \cdot \nabla_{\theta} J_i(\theta) \tag{4}$$

In summary, we can describe the SGD algorithm in Fig. 2.

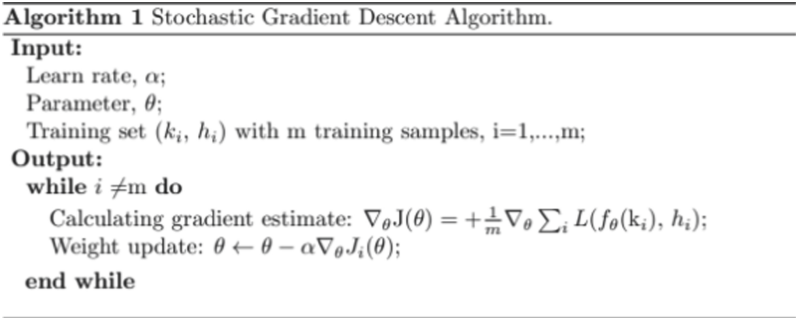


Fig. 2. The algorithm of stochastic gradient descent (SGD)

3.3 Methodology of Semantic Segmentation

We apply a combination of VGG19 and SegNet as a semantic segmentation model and employ our datasets to train and test the model, our dataset consists of 91 images as shown in Fig. 3. This paper reflects each image at a 50% probability level through data



Fig. 3. The dataset of semantic segmentation

augmentation. The translation is performed in the unit of pixels from the horizontal and vertical directions, the translation is randomly selected from the continuous uniform distribution within the interval $[-10, 10]$.

The structure of our neural network proposed in this paper utilizes a combination of encoders and decoders to produce feature maps of images. The encoder of SegNet includes convolutional layer, batch normalization, ReLU activation, and max pooling. Max pooling is used for reducing the size of feature maps. Even though the object boundary in the image may be blurred during the operation of max pooling, the pooling is indeed the best way to reduce the size of the feature maps. For reducing the feature map size while retaining the complete boundary information, SegNet extracts boundary information from the feature maps before performing the downsampling. During the decoding process, the upsampling operation of the decoder preserves the size of the original input. The max pooling memory index stored in each encoder map is used for upsampling the feature map. The last decoder is connected to the softmax classifier so as to assign the label of each class for the image [16].

Assume a is an array received from the upper layer, a_k is the k -th element in the array, and i is the total number of the array [17]

$$\text{soft max}(k) = \frac{\exp(a_k)}{\sum_i \exp(a_i)}. \quad (5)$$

This paper created a SegNet network whose weight was initialized from the VGG-19 network. The additional layer required for semantic segmentation replaces the last pooling layer.

In summary, our semantic segmentation model takes advantage of the encoder-decoder structure, which combines visual information with high-level features through pooling to understand the surrounding scene in detail.

3.4 Methodology of Vehicle Detection

Our vehicle detector is a Faster RCNN-based model as shown in Fig. 4, which uses 337 images from our traffic scenes as datasets as shown in Fig. 5. The size of the input layer is 32×32 for better convolution operations. All convolutional layers of this model use a 3×3 convolution kernel, we set the step size as 1. Moreover, in the convolutional layer of the Faster R-CNN, all the convolutions are subject to one padding expansion, resulting in an increase of 2 in length and width. This setup does not change the size of the inputs and outputs.

Similarly, the convolution kernel size and step size of the pooling layer in the model are both set to 2. Thus, the size of each matrix passing through the pooling layer becomes one-half of the original. In other words, the convolutional layer and the ReLU layer maintain the size of the feature maps, the pooling layer reduces the size of the feature maps to 0.25.

Moreover, this model converts the collected proposals in the RoI Pooling layer into 7×7 proposal feature maps and sends them to the classification layer. Feature maps are classified using the cross-entropy as a loss function in the classification layer.

By adjusting a series of network parameters, vehicle detector can better learn semantics from shallow to deep and learn feature maps from abstract to specific.

Moreover, Faster R-CNN has two fully connected layers for classification and regression, respectively. Similarly, the most usage of two loss functions is fine-tuning. If i is assumed to be the index of an anchor in a mini-batch, p_i is the probability that the algorithm output, p_i^* is the ground truthing label of the anchor i , t_i is a four-element vector, which is bounded by the algorithm. The parameterized coordinates of the box t_i^* are the ground truthing box associated with a positive anchor [18],

$$L(\{p_i\}, \{t_i\}) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*) \quad (6)$$

where $L_{cls}(p_i, p_i^*)$ is the logarithmic loss of the classification

$$L_{cls}(p_i, p_i^*) = -\log[p_i p_i^* + (1 - p_i^*)(1 - p_i)] \quad (7)$$

The classification calculates logarithm loss for each anchor, which is summed and divided by the total number of anchors N_{cls} .

In the regression loss,

$$L_{reg}(t, t_i^*) = R(t_i - t_i^*) \quad (8)$$

where R is defined as smooth L_1 and σ is 3, $x = t_i - t_i^*$,

$$R = smoothL_1(x) = \begin{cases} 0.5x^2 \times 1/\sigma^2 & \text{if } |x| < 1/\sigma^2 \\ |x| - 0.5 & \text{otherwise.} \end{cases} \quad (9)$$

For each anchor, we calculate $L_{reg}(t_i, t_i^*)$ and multiply it by p^* , then sum and multiply it by using a factor λ/N_{reg} . p^* has an object (+1) and no object (-1), which means that only the foreground is used to calculate the cost, the background does not calculate the loss. N_{reg} is the size of feature maps, λ controls the weight for classification and regression at a stable level.

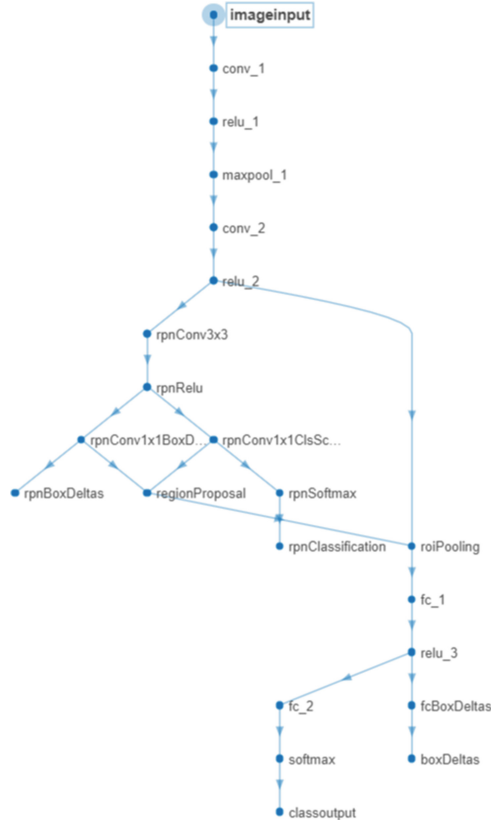


Fig. 4. The network structure of vehicle detector

4 Results

We use deep learning to achieve scene understanding. In order to verify the advantages of deep learning, we explore semantic segmentation and vehicle detection through experiments. As shown in Fig. 6, visually, the model in the semantic segmentation is satisfactory for the segmentation of buildings, sky, bus lane, and vehicles.

However, visual objects such as pedestrians, lanes, and trees are not accurate. This paper uses IoU to measure the amount of overlap for each class. Taken the first segmentation result in Fig. 6, the IoU of each category shows that the building IoU is up to 89% with the highest accuracy among the 11 classes. At the same time, the

proposed model scored higher accuracy on roads, sky, bus lane, and vehicles with 66%, 81%, 86%, and 66% respectively. However, the scores for segmentations of lane, tree, and turning sign are lower. The reason why we get these results is that the objects in the dataset lane and traffic sign are relatively smaller, the possibility of occurrence is rare. In future, we will focus on collecting the visual data including lane, tree, and turning sign images.



Fig. 5. The dataset of vehicle detection

For each class, the accuracy is the ratio of correctly categorized pixels to the total number of pixels in that class. The accuracy of sky detection is the highest among all classes at 91%. Secondly, bus lane, road, lane, and building classes have achieved satisfactory levels of accuracy 90%, 86%, 70%, and 81% respectively. Meanwhile, in this paper we exploited IoU to measure the overlapping between the ground truth and the detected regions, generated the results from the proposed model. The overlapping rates of buildings, sky, and bus lane are 71%, 74%, and 76%, respectively. In contrast, the accuracy and IoU of lane detection are worse than those of other classes, because the labelled area is small, its occurrence is lower than other classes.

From the experimental results of vehicle detection, the detector can successfully detect vehicles in multiple directions, sizes, and types under normal conditions, even detect vehicles with only a half of the vehicle appeared in the imaging range.

However, if there is an interference in the detection environment, our detection accuracy will be decreased. When a road tax sticker appears on the car windscreen, the detector incorrectly detects the road tax sticker as a vehicle. On the other hand, the detector can roughly detect the presence or absence of a vehicle in a particular area, but the position of the label boxes is offset.

This paper takes advantage of average precision to measure the accuracy of the detector. Both precision and recall are based on an understanding and measure of relevance. The best training result of average precision for this paper is 81%, which is based on a 22-layer Faster R-CNN network.

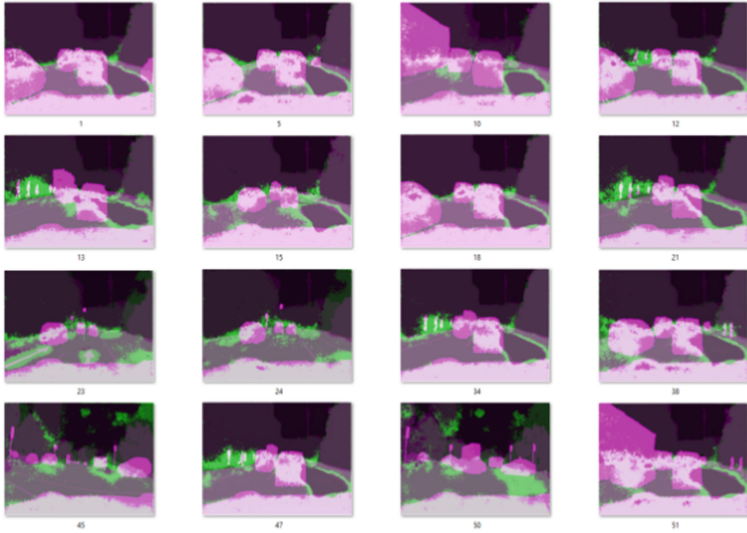


Fig. 6. The segmentation results

The log-average miss rate of the detection results is compared to the ground truth, which is adopted to measure the performance of the object detector in this paper, miss rate decreases as the false positive per image (FPPI) grows, and the log-average miss rate is 0.4.

In the vehicle identification, the test results were up to 81%, the average error rate was as low as 0.4. As shown in Fig. 7, compared with several models and models of different deep networks of the same class, this model is better in the control of the position and size of the bounding box.

Table 1. Comparison of VGG16-SegNet and VGG19-SegNet in IoU

Class	VGG16-SegNet IoU	VGG19-SegNet IoU
Sky	66%	74%
Building	70%	71%
Buslane	59%	76%
Road	59%	61%
Lane	50%	50%
Vehicle	67%	60%



Fig. 7. Comparison of four models with the same image

In the semantic segmentation, multiple evaluation indicators are taken as the metrics. We compare the network performance using VGG16 and VGG19 as the basic models shown in Table 1, respectively. The results of IoU show that segmentation results of VGG19 on Sky and Buslane are significantly higher than VGG16-SegNet. VGG16 is higher than VGG19 only in the segmentation result of the vehicle. After measuring all the evaluation indicators in general, we choose VGG19 as the basic model for our segmentation.

Through the exploration of vehicle detection and semantic segmentation, we find that deep learning has a great positive effect on scene understanding which relies on a layered processing mechanism and powerful transfer capabilities to improve the performance of the model in scene understanding.

5 Conclusion

The goal of this paper is to achieve traffic scene understanding using deep learning including semantic segmentation and vehicle detection. We fulfil each phase of the paper such as dataset preprocessing, neural network design and training, model evaluation, resultant comparisons. This paper provides a large number of original images and annotated images of our traffic environment for neural network computations, including annotations for image segmentation and vehicle identification. Furthermore,

SegNet shows high accuracy with small datasets. Faster R-CNN adopts two sets of convolution units to achieve high-precision segmentation and recognition. In future, we will use ensemble learning to integrate our experimental results together so as to get better results [24–27].

References

1. Li, Y., Dong, G., Yang, J., Zhang, L., Gao, S.: 3D point cloud scene data acquisition and its key technologies for scene understanding. *Laser Optoelectron. Progr.* **56**(4), 040002 (2019)
2. Chen, H., et al.: The rise of deep learning in drug discovery. *Drug Discov. Today* **23**(6), 1241–1250 (2019)
3. Husain, F., Dellen, B., Torras, C.: Scene understanding using deep learning, pp. 373–382. Academic Press (2017)
4. Yang, S., Wang, W., Liu, C., Deng, W.: Scene understanding in deep learning-based end-to-end controllers for autonomous vehicles. *IEEE Trans. Syst. Man Cybern.: Syst.* **49**(1), 53–63 (2019)
5. Jin, Y., Li, J., Ma, D., Guo, X., Yu, H.: A semi-automatic annotation technology for traffic scene image labelling based on deep learning pre-processing. In: *IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, pp. 315–320 (2017)
6. Nikita, D., Konstantin, S., Julien, M., Cordelia, S.: BlitzNet: A real-time deep network for scene understanding. In: *IEEE International Conference on Computer Vision (ICCV)*, pp. 4154–4162 (2017)
7. Yao, W., Zeng, Q., Lin, Y., Guillemard, F., Geronimi, S., Aioun, F.: On-road vehicle trajectory collection and scene-based lane change analysis. *IEEE Trans. Intell. Transp. Syst.* **18**(1), 206–220 (2017)
8. Wei, Y., Tian, Q., Guo, J., Huang, W., Cao, J.: Multi-vehicle detection algorithm through combining Harr and HOG features. *Math. Comput. Simul.* **155**, 130–145 (2017)
9. Lecun, Y., Muller, U., Ben, J., Cosatto, E., Flepp, B.: Off-road obstacle avoidance through end-to-end learning. In: *International Conference on Neural Information Processing Systems*, pp. 739–746 (2005)
10. Ohsugi, H., Tabuchi, H., Enno, H., Ishitobi, N.: Accuracy of deep learning, a machine-learning technology using ultra-wide-field fundus ophthalmoscopy for detecting hematogenous retinal detachment. *Sci. Rep.* **7**(1), 9425 (2017)
11. Li, F., Deng, J., Li, K.: ImageNet: constructing a largescale image database. *J. Vis.* **9**(8), 1037–1038 (2009)
12. Samui, P., Roy, S.S., Balas, V.E.: *Handbook of Neural Computation*, pp. 12–34. Academic Press, Cambridge (2017)
13. Yu, Y., Cao, K.: A method for semantic representation of dynamic events in traffic scenes. *Inf. Control* **44**(1), 83–90 (2015)
14. Newton, A., Pasupathy, R., Yousefian, F.: Recent trends in stochastics gradient descent for machine learning and big data. In: *Winter Simulation Conference (WSC)*, pp. 366–380 (2018)
15. Goodfellow, I., Bengio, Y., Courville, A.: *Deep Learning*, pp. 44–56. The MIT Press, Cambridge (2016)
16. Tran, S., Kwon, O., Kwon, K., Lee, S., Kang, K.: Blood cell images segmentation using deep learning semantic segmentation. In: *IEEE International Conference on Electronics and Communication Engineering (ICECE)*, pp. 13–16 (2018)

17. Badrinarayanan, V., Handa, A., Cipolla, R.: SegNet: a deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(12), 2481–2495 (2017)
18. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. *Pattern Anal. Mach. Intell.* **39**(6), 1137–1149 (2017)
19. Ji, H., Liu, Z., Yan, W., Klette, R.: Early diagnosis of Alzheimer’s disease based on selective kernel network with spatial attention. In: *ACPR* (2019)
20. Al-Sarayreh, M., Reis, M.M., Yan, W.Q., Klette, R.: A sequential CNN approach for foreign object detection in hyperspectral images. In: Vento, M., Percannella, G. (eds.) *CAIP 2019*. LNCS, vol. 11678, pp. 271–283. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-29888-3_22
21. Ji, H., Liu, Z., Yan, W., Klette, R.: Early diagnosis of Alzheimer’s disease using deep learning. In: *ICCCV 2019*, pp. 87–91 (2019)
22. Shen, Y., Yan, W.: Blindspot monitoring using deep learning. In: *IVCNZ* (2018)
23. Pan, C., Li, X., Yan, W.: A learning-based positive feedback approach in salient object detection. In: *IVCNZ* (2018)
24. Wang, X., Yan, W.Q.: Cross-view gait recognition through ensemble learning. *Neural Comput. Appl.* (2019). <https://doi.org/10.1007/s00521-019-04256-z>
25. Liu, X.: Vehicle-related Scene Understanding. Masters thesis, Auckland University of Technology, New Zealand (2019)
26. Wang, X., Yan, W.: Human gait recognition based on frame-by-frame gait energy images and convolutional long short-term memory. *Int. J. Neural Syst.* **30**(1), 1950027:1–1950027:12 (2020)
27. Wang, X., Zhang, J., Yan, W.Q.: Gait recognition using multichannel convolution neural networks. *Neural Comput. Appl.* (2019). <https://doi.org/10.1007/s00521-019-04524-y>



Spatiotemporal Saliency Based Multi-stream Networks for Action Recognition

Zhenbing Liu^{1,2,3}, Zeya Li^{1,2,3}, Ming Zong^{1,2,3(✉)}, Wanting Ji^{1,2,3},
Ruili Wang^{1,2,3}, and Yan Tian^{1,2,3}

¹ School of Computer Science and Information Security,
Guilin University of Electronic Technology, Guilin, China

² School of Natural and Computational Sciences, Massey University,
Auckland, New Zealand

M. Zong@massey.ac.nz

³ School of Computer and Information Engineering,
Zhejiang Gongshang University, Hangzhou, China

Abstract. Human action recognition is a challenging research topic since videos often contain clutter backgrounds, which impairs the performance of human action recognition. In this paper, we propose a novel spatiotemporal saliency based multi-stream ResNet for human action recognition, which combines three different streams: a spatial stream with RGB frames as input, a temporal stream with optical flow frames as input, and a spatiotemporal saliency stream with spatiotemporal saliency maps as input. The spatiotemporal saliency stream is responsible for capturing the spatiotemporal object foreground information from spatiotemporal saliency maps which are generated by a geodesic distance based video segmentation method. Such architecture can reduce the background interference in videos and provide the spatiotemporal object foreground information for human action recognition. Experimental results on UCF101 and HMDB51 datasets demonstrate that the complementary spatiotemporal information can further improve the performance of action recognition, and our proposed method obtains the competitive performance compared with the state-of-the-art methods.

Keywords: Action recognition · Spatiotemporal saliency map image · ResNet

1 Introduction

Human action recognition is a process of labeling video frames with action labels [1, 3, 27, 29]. It has a wide range of applications in real life such as intelligent surveillance, virtual reality (VR), video retrieval, intelligent human-computer interaction and shopping behavior analysis.

Conventional human action recognition methods based on handcrafted features cannot fully extract efficient and robust features from videos, especially when there are complex clutter backgrounds in the videos such as target occlusion, illumination variation, and camera movement. To address this, deep convolutional neural network (ConvNets) based human action recognition methods have been developed, which can

be categorized into two categories: two-stream convolutional neural networks based methods [1–3] and 3D convolutional neural networks based methods [5–7]. Typically, a two-stream convolutional neural network consists of two streams: a spatial stream and a temporal stream. The spatial stream is used to capture the appearance information from a video, while the temporal stream is used to capture the motion information from the video. Different from two-stream convolutional neural networks, 3D convolutional neural networks can simultaneously learn the spatial and temporal information from multiple consecutive video frames.

For the accuracy of human action recognition, the clutter backgrounds impose a negative effect. To solve this problem, we propose a novel spatiotemporal saliency based multi-stream ResNet (STS multi-stream model) for human action recognition, which combines three different streams including a spatial stream, a temporal stream, and a spatiotemporal saliency stream. Given a video, the spatial stream utilizes the RGB frames of the video as input, and the temporal stream utilizes the optical flow frames of the video as input. The spatiotemporal saliency maps, which is obtained by a geodesic distance based video segmentation method [14], is used as the input of the spatiotemporal saliency stream. This can capture the spatiotemporal object foreground information in the video and suppress the background information.

The contributions of this paper include: (i) We propose a novel spatiotemporal saliency based multi-stream ResNet for human action recognition, which consists of a spatial stream, a temporal stream and a spatiotemporal stream. (ii) The novel spatiotemporal saliency stream can reduce the background interference in videos and provide the spatiotemporal object foreground information for human action recognition. (iii) We propose an averaging fusion for the outputs of the three streams.

The rest of this paper is organized as follows. Section 2 presents related work. The proposed method is detailed in Sect. 3. Section 4 shows the results of conducted extensive experiments. Section 5 provides the conclusions of the paper.

2 Related Work

2.1 Two-Stream Based 2D Convolutional Neural Networks

Recently, two-stream based 2D convolutional neural networks are widely applied for human action recognition. Simonyan *et al.* [1] first proposed a two-stream ConvNet architecture, in which spatial and temporal neural networks were developed to capture spatial and temporal information of videos separately, and the output of these two networks were combined by late fusion. Wang *et al.* [2] proposed the temporal segment network (TSN) with four types of input modalities, which was based on the idea of long-range temporal video structure modeling. Feichtenhofer *et al.* [3] proposed spatiotemporal residual networks (ST-ResNet) to add residual connections between different layers and learned spatiotemporal features by connecting the appearance channel and motion channel. Wang *et al.* [4] developed a spatiotemporal pyramid network to fuse the spatial and temporal features. A spatiotemporal compact bilinear operator was adopted to enable unified modeling of various fusion strategies. Jing *et al.* [12] combined multiple streams with dynamic images, optical flow frames and raw frames as

input to improve the performance of action recognition. Liu *et al.* [13] proposed a multi-stream neural network by using RGB frames, dense optical flow frames and gradient maps as the input, where different streams were responsible for capturing various appearance and motion feature information.

2.2 3D Convolutional Neural Networks (3D CNNs) and Others

Since 3D convolution can process multiple consecutive images at the same time, 3D convolution neural networks have the ability to extract temporal information between video frames. Ji *et al.* [5] firstly developed a 3D CNN model that provided multiple channels from adjacent input frames and performed 3D convolution for each channel. Tran *et al.* [6] proposed Convolutional 3D (C3D) which used multi-frames as an input of the network. Diba *et al.* [7] developed Temporal 3D ConvNets (T3D) by deploying a 3D temporal transition layer (TTL) instead of a transition layer in DenseNet [28]. Qiu *et al.* [8] developed a residual learning model by using different convolution filters and proposed the Pseudo-3D Residual Net (P3D ResNet). However, 3D CNNs based networks need training much more parameters and cost expensive computation compared with 2D CNNs based networks [6].

In addition to the development of two-stream networks and 3D CNNs, some research contributes to the related fields (such as data input, model architecture, and fusion) to address the challenges in human action recognition. Kar *et al.* [9] developed AdaScan to dynamically pool the key informative frames and proposed a pooled feature vector for human action recognition. Sun *et al.* [10] proposed a compact motion representation which can be embedded in any existing CNN based video action recognition framework with a slight additional cost. Xie *et al.* [11] combined top-heavy model design, temporally separable convolution, and spatiotemporal feature gating together to improve the performance of action recognition.

2.3 Residual Network

The deep residual network has obtained a good performance in image recognition [16]. Different from deep neural networks using multiple stacked layers $F(x)$ to approximate the desired underlying mapping $H(x)$, residual networks consider using multiple stacked layers $F(x)$ to approximate a residual mapping $H(x)-x$. Figure 1 illustrates the basic residual building block.

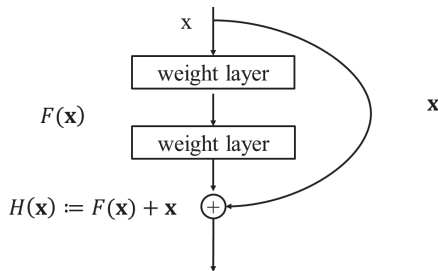


Fig. 1. The basic residual building block.

Hara *et al.* [24] proposed a 3D CNN based residual network which was made up of residual blocks with 3D convolutions and 3D pooling to extract spatiotemporal features. Lei *et al.* [25] proposed a temporal deformable ResNet to analyze the most suitable sampling intervals. Compared with conventional convolution neural networks, deep residual networks add shortcut connections from the front convolution layers to the later convolution layers. This can bypass the intermediate layers and propagate information to the later layers directly [21]. By conducting experiments with different numbers of layers, a 101-layer ResNet is chosen as the backbone network.

3 Proposed Method

In this section, we first introduce the spatiotemporal saliency map generated by [14]. Then we present the proposed spatiotemporal saliency based multi-stream ResNet for human action recognition.

3.1 Spatiotemporal Saliency Map

The generation of spatiotemporal saliency map is based on a geodesic distance based video segmentation method [14], which can distinguish the foreground object and surrounded background areas by the corresponding spatiotemporal edge values. The procedure can be summarized as the following steps:

- (i) Obtaining a superpixel set for the input video frames by using k-means clustering method [15];
- (ii) Obtaining a spatial edge probability map by using edge detection approach [22];
- (iii) Obtaining the temporal gradient magnitude of the optical flow frames [17];
- (iv) Computing the spatial edge probability of each superpixel to obtain the spatial superpixel edge maps;
- (v) Computing the temporal gradient magnitude of each superpixel to obtain the temporal superpixel optical flow magnitude map;
- (vi) Obtaining the spatiotemporal edge probability map by combining the spatial superpixel edge map and the temporal superpixel optical flow magnitude map;
- (vii) Obtaining the spatiotemporal saliency map from the spatiotemporal edge probability map by calculating the probability of foreground object based on the geodesic distance.

The results of the generated spatiotemporal saliency map by geodesic distance based video segmentation method can be shown in Fig. 2. The spatiotemporal saliency map contains human object foreground information and edge information which can provide more prior spatiotemporal knowledge for human action recognition.

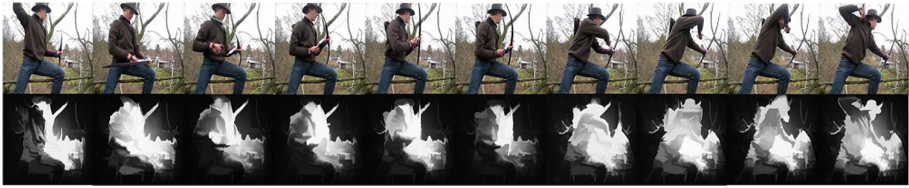


Fig. 2. Spatiotemporal saliency maps generated by geodesic distance based video object segmentation. The top row shows 10 consecutive RGB frames sampled with a fixed time interval in the Archery video from UCF101 dataset [18], and the second row illustrates the corresponding spatiotemporal saliency maps.

3.2 STS Multi-stream Model

Architecture Framework

The architecture framework of our proposed spatiotemporal saliency based multi-stream ResNet (STS multi-stream model) is illustrated in Fig. 3. It consists of three streams with the input of RGB frames, optical flow frames, and spatiotemporal saliency maps respectively. The spatial stream is responsible for capturing the appearance information from raw RGB frames, the temporal stream is responsible for capturing the motion information from optical flow frames, and the spatiotemporal saliency stream is responsible for capturing the spatiotemporal object foreground information from spatiotemporal saliency maps. The neural networks for the spatial stream, the temporal stream, and the spatiotemporal saliency stream are trained individually. Finally, the outputs of the softmax layers of the three streams are averaging fused to form a final softmax score for human action recognition.

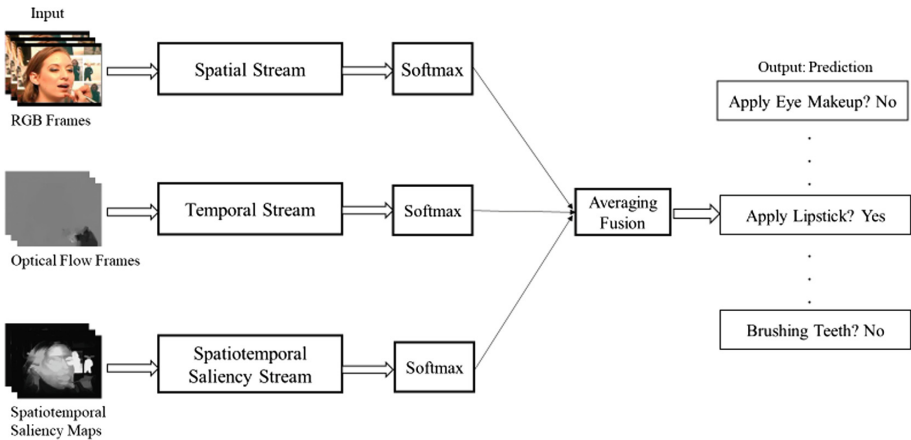


Fig. 3. The architecture framework of our proposed spatiotemporal saliency based multi-stream ResNet for human action recognition. It consists of a spatial stream with RGB frames as input, a temporal stream with optical flow frames as input, and a spatiotemporal saliency stream as input.

Training of Three Streams

The proposed spatiotemporal saliency based multi-stream ResNet consists of three different streams: a spatial stream, a temporal stream, and a spatiotemporal saliency network. We train the three streams separately to extract the appearance information, motion information and spatiotemporal saliency information from videos. All training details are summarized as follows.

Spatial Stream. The spatial stream with RGB frames as input provides the basic appearance characteristics of the video, which is the most important stream in the action recognition process [2].

The input of the spatial stream consists of multiple RGB frames obtained in a random sampling interval from the extracted video frames. Similar to the temporal segment network [2] training strategy, we randomly select three video frames from a video for representing the video. Then a consensus among the selected frames is derived as the video-level prediction. We input the three video frames separately into the spatial stream and calculate the losses individually, then these losses will be added as the final loss for backpropagation. The output of the softmax layer represents the output of the spatial stream for this video.

Temporal Stream. The temporal stream with optical flow frames as input provides the motion information of the action, which has been crucial for action recognition.

We use the Optical Flow Estimation [17] method to obtain optical flow frames from the raw RGB frames of videos. Different from the input of the spatial stream, we randomly select a series of stacked optical flow frames from the optical flow frames as the input of the temporal stream. The outputs of the softmax layer represent the output of the temporal stream.

Spatiotemporal Saliency Stream. The spatiotemporal saliency stream with spatiotemporal saliency maps as input provides the spatiotemporal object foreground information and reduce the background interference.

We utilize a geodesic distance based video segmentation method [14] to obtain the spatiotemporal saliency maps from the RGB frames and optical flow frames. Similar to the input of the spatial stream, we randomly select five frames from the spatiotemporal saliency maps, and we input the five spatiotemporal saliency maps separately into the spatiotemporal saliency stream and calculate the losses individually. Then these losses will be added as the final loss for backpropagation. The output of the softmax layer represents the output of the spatiotemporal saliency stream.

Fusion of Three Streams. In order to verify the proposed multi-stream model, we combine the outputs of all three streams to fuse spatial information and temporal information of a video.

In the process of training the proposed multi-stream model, we use cross-entropy loss uniformly in each stream. The loss function is shown in Eq. (1).

$$H_{y'}(y) = - \sum_i y'_i \log(y_i) \quad (1)$$

where y' represents the prediction result, y represents the target label, and i denotes the i th input image index.

After the ResNets in the three streams are trained separately, each stream can extract the spatial features, motion features, spatiotemporal saliency features separately. The final output prediction is fused with the weighted outputs of the softmax layers of all the streams.

4 Experiments

4.1 Datasets

We evaluate the performance of our proposed STS multi-stream model on UCF-101 [18] and HMDB [23] datasets. The UCF-101 dataset consists of 101 action categories with 13320 video clips. The HMDB-51 dataset includes 6849 video clips divided into 51 action categories, and each category contains a minimum of 101 video clips. We use the pre-provided training/test split of the UCF-101, which divides the UCF-101 dataset into 9537 training videos and 3783 testing videos. Similarly, we use the pre-provided training/test split of the HMDB-51, which contains about 3750 training videos and 3099 test videos.

4.2 Learning Process

We use Pytorch to implement our proposed STS multi-stream model and train the model on 4 Nvidia GTX 1080Ti GPUs. We set the learning rate to 0.001 and use a mini-batch size of 32. We adopt 101-layer ResNet (ResNet-101 for short) for the spatial stream, the temporal stream and the spatiotemporal saliency stream. We first use the pre-trained ResNet-101 on the ImageNet dataset, which is a large-scale hierarchical image database containing more than 1 million images [19], as the spatial stream model parameter initialization. Then we finetune the pre-trained ResNet-101 on the UCF-101 and HMDB-51 datasets. For the temporal stream, by averaging the weight value across RGB channels and replicate this value by the channel number of motion stream input, we use ImageNet pre-trained weights and modify the weights of the first convolution layer pre-trained on ImageNet from (64, 3, 7, 7) to (64, 20, 7, 7), which contains 10 x -channel and 10 y -channel optical flow frames. Similar to the spatial stream, we use the pre-trained ResNet-101 on ImageNet and finetune the spatiotemporal saliency stream.

4.3 Experimental Results

The experimental results are reported in Table 1. It is obvious that the accuracy of the input with two modalities (such as RGB frames + Optical Flow frames) is higher than the input with a single modality (such as RGB frames) on both UCF-101 dataset and HMDB-51 datasets. Further, we can find that the input with optical flow frames and spatiotemporal saliency improves 0.7% and 2.2% than the input with only optical flow frames on UCF-101 and HMDB-51 datasets, respectively. The addition of spatiotemporal saliency stream can provide the spatiotemporal object foreground information and

reduce the background interference, which is beneficial for action recognition. A similar phenomenon can be verified when we use RGB frames and spatiotemporal saliency maps as input, the input with RGB frames and spatiotemporal saliency improves 1.2% and 2.6% than the input with only RGB frames on UCF-101 and HMDB-51 datasets, respectively. When we fuse all these three streams, we can obtain the best accuracy of 90.1% and 62.4% on UCF-101 and HMDB-51 datasets, respectively. The input with all three modalities improves 2.9% and 1.9% than the input with RGB frames and optical flow frames on UCF-101 and HMDB-51 datasets, respectively, which demonstrates that the spatiotemporal saliency stream can further provide effective supplementary information for improving the performance of action recognition.

Table 2 compares the experimental results of the proposed STS multi-stream method and other state-of-the-art methods for human action recognition. The proposed STS multi-stream model is superior to iDT [20], Two-stream [1], Two-stream + LSTM [26], C3D [6] and RGB+OF+DI with 3D CNN [11]. Especially compared with other two-stream based models such as Two-stream [1] and Two-stream + LSTM [26], our proposed STS multi-stream model obtains better performance since the spatiotemporal saliency stream can provide the spatiotemporal object foreground information and reduce the background interference.

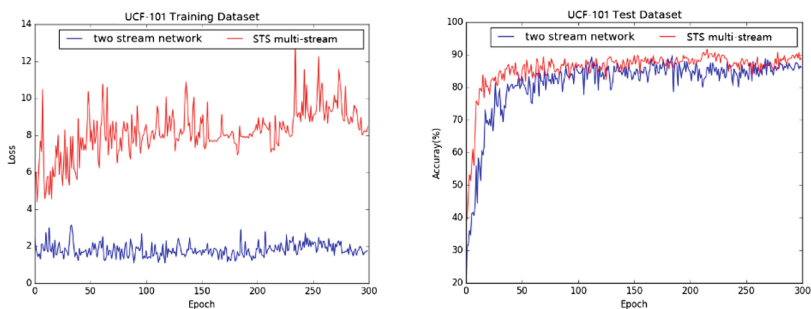


Fig. 4. The loss scores and accuracies of two comparison methods on the UCF-101 dataset.

The loss scores and classification accuracies of two stream method and our STS multi-stream method for human action recognition are illustrated in Fig. 4. As shown in Fig. 4, the proposed fusion stream obtains better performance than two stream method.

Table 1. The accuracy of different modalities on the UCF-101 and HMDB-51 datasets.

Input	UCF-101	HMDB-51
RGB	81.3%	50.1%
Optical Flow	79.7%	55.6%
RGB + Optical Flow	87.2%	60.5%
RGB + Spatiotemporal Saliency	82.5%	53.7%
Optical Flow + Spatiotemporal Saliency	80.4%	57.8%
RGB + Optical Flow + Spatiotemporal Saliency	90.1%	62.4%

Table 2. Comparison of our method based on multi-stream with the state-of-the-art methods on the UCF101 and HMDB51 datasets.

Algorithm	UCF-101	HMDB-51
iDT [20]	86.4%	57.2%
Two-stream [1]	88.0%	59.4%
Two-stream + LSTM [26]	88.6%	–
C3D [6]	85.2%	–
RGB + OF + DI with 3D CNN [11]	88.6%	57.9%
STS multi-stream model	90.1%	62.4%

5 Conclusion

In this paper, we propose a novel spatiotemporal saliency based multi-stream ResNet for human action recognition, which combines three different streams: a spatial stream with RGB frames as input, a temporal stream with optical flow frames as input and, a spatiotemporal saliency stream with spatiotemporal saliency maps as input. Compared with conventional two-stream based models, the proposed method can provide the spatiotemporal object foreground information and reduce the background interference, which has been verified effective for human action recognition. Experimental results demonstrate that our proposed STS multi-stream model achieves the best accuracy compared with the input with single modality or two modalities. In the future, we will further explore sharing information between different streams to improve the performance of human action recognition.

Acknowledgment. This study is supported by the National Natural Science Foundation of China (Grant No. 61562013), the Natural Science Foundation of Guangxi Province (CN) (2017GXNSFDA198025), the Study Abroad Program for Graduate Student of Guilin University of Electronic Technology (GDYX2018006), the Marsden Fund of New Zealand, the National Natural Science Foundation of China (Grant 61602407), Natural Science Foundation of Zhejiang Province (Grant LY18F020008), the China Scholarship Council (CSC) and the New Zealand China Doctoral Research Scholarships Program.

References

1. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Advances in Neural Information Processing Systems (NIPS) (2014)
2. Wang, L., et al.: Temporal segment networks for action recognition in videos. *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(11), 2740–2755 (2018)
3. Feichtenhofer, C., Pinz, A., Wildes, R.P.: Spatiotemporal residual networks for video action recognition. In: Advances in Neural Information Processing Systems (NIPS) (2016)
4. Wang, Y., et al.: Spatiotemporal pyramid network for video action recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
5. Ji, S., et al.: 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(1), 221–231 (2013)

6. Tran, D., et al.: Learning spatiotemporal features with 3D convolutional networks. In: IEEE International Conference on Computer Vision (ICCV) (2014)
7. Diba, A., et al.: Temporal 3D ConvNets: new architecture and transfer learning for video classification. arXiv preprint [arXiv:1711.08200](https://arxiv.org/abs/1711.08200) (2017)
8. Qiu, Z., Yao, T., Mei, T.: Learning spatio-temporal representation with pseudo-3D residual networks. In: IEEE International Conference on Computer Vision (ICCV) (2017)
9. Kar, A., et al.: AdaScan: adaptive scan pooling in deep convolutional neural networks for human action recognition in videos. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
10. Sun, S., et al.: Optical flow guided feature: a fast and robust motion representation for video action recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
11. Xie, S., et al.: Rethinking spatiotemporal feature learning: speed-accuracy trade-offs in video classification. In: European Conference on Computer Vision (ECCV) (2017)
12. Jing, L., Ye, Y., Yang, X., Tian, Y.: 3D convolutional neural network with multi-model framework for action recognition. In IEEE International Conference on Image Processing (ICIP) (2017)
13. Liu, X., Yang, X.: Multi-stream with deep convolutional neural networks for human action recognition in videos. In: Cheng, L., Leung, A.C.S., Ozawa, S. (eds.) ICONIP 2018. LNCS, vol. 11301, pp. 251–262. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-04167-0_23
14. Wang, W., Shen, J., Porikli, F.: Saliency-aware geodesic video object segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
15. Achanta, R., et al.: SLIC superpixels compared to state-of-the-art superpixel methods. IEEE Trans. Pattern Anal. Mach. Intell. **34**(11), 2274–2281 (2012)
16. He, K., et al.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
17. Brox, T., Bruhn, A., Papenbergh, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: Pajdla, T., Matas, J. (eds.) ECCV 2004. LNCS, vol. 3024, pp. 25–36. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24673-2_3
18. Soomro, K., Zamir, A.R., Shah, M.: UCF101: a dataset of 101 human actions classes from videos in the wild. arXiv preprint [arXiv:1212.0402](https://arxiv.org/abs/1212.0402) (2012)
19. Deng, J., et al.: ImageNet: a large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2009)
20. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: IEEE International Conference on Computer Vision (ICCV) (2014)
21. Sun, L., et al.: Lattice long short-term memory for human action recognition. In: IEEE International Conference on Computer Vision (ICCV) (2017)
22. Leordeanu, M., Sukthankar, R., Sminchisescu, C.: Efficient closed-form solution to generalized boundary detection. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7575, pp. 516–529. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33765-9_37
23. Kuehne, H., et al.: HMDB51: a large video database for human motion recognition. In: IEEE International Conference on Computer Vision (ICCV) (2013)
24. Hara, K., Kataoka, H., Satoh, Y.: Learning spatio-temporal features with 3D residual networks for action recognition. In: IEEE International Conference on Computer Vision (ICCV) (2017)
25. Lei, P., Todorovic, S.: Temporal deformable residual networks for action segmentation in videos. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)

26. Ng, Y.H., et al.: Beyond short snippets: deep networks for video classification. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
27. Poppe, R.: A survey on vision-based human action recognition. *Image Vis. Comput.* **28**(6), 976–990 (2010)
28. Huang, G., et al.: Densely connected convolutional networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
29. Tian, C., et al.: Image denoising using deep CNN with batch renormalization. *Neural Netw.* **121**, 461–473 (2020)



Attention Guided Unsupervised Image-to-Image Translation with Progressively Growing Strategy

Yuchen Wu¹(✉) , Runtong Zhang¹(✉) , and Keiji Yanai²(✉) 

¹ University of Electronic Science and Technology of China, Chengdu, China
1234567890@gmail.com, 3313560262@qq.com

² The University of Electro-Communications, Tokyo, Japan
yanai@cs.uec.ac.jp

Abstract. Unsupervised image-to-image translation such as CycleGAN has received considerable attention in recent research. However, when handling large images, the quality of generated images are not in good quality. Progressive Growing GAN has proved that progressively growing of GANs could generate high pixels images. However, if we simply combine PG-method and CycleGAN, it must bring model collapse. In this paper, motivated from skip connection, we propose Progressive Growing CycleGAN (PG-Att-CycleGAN), which can stably grow the input size of both the generator and discriminator progressively from 256×256 to 512×512 and finally 1024×1024 using the weight α . The whole process makes generated images clearer and stabilizes training of the network. In addition, our new generator and discriminator cannot only make the domain transfer more natural, but also increase the stability of training by using the attention block. Finally, through our model, we can process high scale images with good qualities. We use VGG16 network to evaluate domain transfer ability.

Keywords: Cycle-Consistent Generative Adversarial Networks · Skip connection · Attention block · Progressive growing strategy

1 Introduction

CycleGAN [1] makes a big progress in unpaired domain translation, which is useful in industrial such as person re-identification [3] and video re-targeting [4]. Larger size pictures are appealing to all of them. With the development of the high-tech camera, there are more and more high pixels images existed. It will be a trend to do domain translation on large size pictures (1024×1024 pixels).

Progressive Growing GAN (PG-GAN) [5] presents progressive growing methods for GANs to process large images, but if we simply cite the progressive growing method in CycleGAN, increasing the layers progressively. However, the generated images are not in good quality, which is shown in Fig. 1. This is because the span of the receptive field is enormous between the layers of CycleGAN,

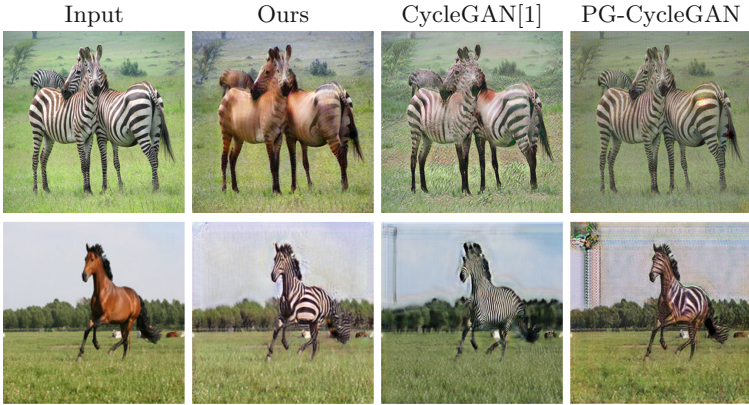


Fig. 1. The result of our GAN, CycleGAN [1], PG-CycleGAN (progressively growing the layer of original CycleGAN), for translating a zebra with a size of 1024×1024 to a horse and a horse to a zebra. By taking the details in the pictures, it is clear to see our generated horse and zebra are more rounded and nature.

which will easily cause the model collapse. Shown in Fig. 1, they only change the color of the whole images, but not the domain.

To prevent such a case, we re-design the generator, whose architecture uses sampling to substitute the stride-2 convolution layers that are used in the original CycleGAN. Besides, the kernel size of the first convolution layers of generators changes from 7×7 to 1×1 to reducing the reconstruction damage caused by encoding and decoding. Moreover, we replace all the transpose convolution layers with bilinear interpolation upsampling layer to erase the checkerboard effect. From Fig. 1, the results from our model have better-translated textures when handling the 1024×1024 size images.

We begin training with the 256×256 size, and after fully trained, we double the size to encourage on fine details. Besides, we use the attention block that protects the high-frequency information to have a clearer image. Comparing with the original CycleGAN and simply growing CycleGAN structure, we qualitatively and quantitatively show that explicitly our new progressive model can do well in domain translation for high pixels pictures.

2 Related Work

2.1 Cycle-Consistent Adversarial Networks

Cycle-Consistent Generative Adversarial Networks (CycleGAN) [1] introduced by Jun-Yan Zhu et al. uses two adversarial processes with two generators and two discriminators to realize two-way domain translation. The key to CycleGAN’s success is the cycle-consistency loss, which represents cycle consistency and guarantees that the learned function can map an individual input to a desired output. The structure of the generator consists of encoders, transformers, and decoders,

which result in a serious problem: edge information will be damaged in the encoding process and cannot be recovered in the decoding process. Therefore, some parts of the generated images are blurry and indistinct. To improve the image quality, we use skip-connection to connect encoder and decoder. Thus the architecture can prevent edge information from being damaged and will be directly transmitted to generated image. Besides, we softly enlarge the generator, which prevents the model collapse. These methods can get better results compared with other models. Finally, our model based on the growing technology can well handle the large scale images.

2.2 Skip Connection

Olaf Ronneberger et al. introduce U-Net [6] to make convolution networks could work for bio-medical images. In terms of the high-quality images, which are important in medical, they use the skip connection between the sampling and upsampling layers. To increase the speed of the architecture, many structures use the sampling to minimize the size of the processing images, which will throw away the high-frequency information that includes the edge information. With the help of the skip connection, the detail information directly transfers to the upsampling layers, thus can have clearer images. We adopt the skip connection between the encoder and decoder inside the generator. Besides, we will also establish a new skip connection with the network growing. Though the network is much deeper, it will still have good quality in generated pictures.

2.3 Progressive Growing of GANs

Progressive Growing of GANs (PG-GANs) [5] realizes size increase by using a progressive growth strategy. In this training process, our model begins from a small output size and gradually adds new layers in output end to expand size, which is realized by weight α changing from 0 to 1. When α increases to 1 as the training process, the new layer is completely added to the model and the output size is expanded. Different from the PG-GAN, our model base on the image input-output structure. Motivated by this progressive growing strategy, we also use weight α to linearly add new layers in both input end and output end to increase image size. Besides, we also increase the size of the discriminator, to prevent model collapse caused by the situation when the discriminator is over trained.

2.4 Artifact

Youssef et al. introduce Attention CycleGAN [7] to protect the background information in datasets like horse2zebra and summer2winter. Using the attention block that only focuses on the domain part, which won't do superfluous translation on the background. Odena et al. [8] discover the checkerboard effect in image processing, which is caused by transpose convolution layers. In this model, we not only present an alternative attention block to keep the milieu but also choose upsampling layers instead of transpose convolution layers to solve the checkerboard effect.

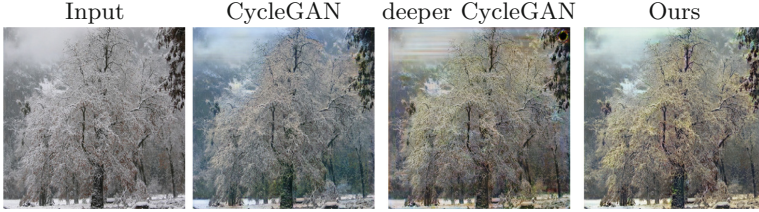


Fig. 2. The result of CycleGAN, deeper CycleGAN (CycleGAN has more layers to deepen the network), our model for translating the picture from winter to summer

3 Proposed Method

Image translation that aims to learn mapping function from source domain to target domain with two sets of independent data, is realized by an Image Transform Net [9]. Style change effect can be improved with more encoder and decoder layers, but the reconstruction loss will increase due to the downsampling process in encoder. Shown in Fig. 2, the deeper CycleGAN can change the color of the tree, while it also causes the sky distorted. To solve this problem, we combine the progressive growth strategy with CycleGAN to propose a new architecture, which can smoothly add new layers to generators after adequate training. Figure 3 visualizes this process.

3.1 Network Structure

Base Structure of Generator. In the generator, the first layer named from-*RGB* is a convolution layer with 1×1 kernel size adopted from PG-GAN [5], which has a good performance on generating high-quality images. The architecture of encoder is adopted from Image Transform Net [9] consisting of two sampling blocks including two 1-stride convolution layers followed by Instance normalization [10] and ReLU, average sampling to shrink images. Same as CycleGAN [1], we use nine residual blocks as the transformer part. Besides, before the transformer part, a skip-connection with weight transmits data skipping the transformer to decoder. For the decoder part, motivated by Stack GAN [11], we choose two bilinear interpolation upsampling layers integrated with two 1-stride convolution layers to expand image size instead of transpose convolution. Moreover, the input of the second upsampling layer is the integration of output from the last layer and data from the first convolution layer in encoder transmitted by a tunnel. Finally, the output is fed in a 1×1 convolution layer named *toRGB* to reduce the dimension back to the RGB image.

Base Structure of Discriminator. We use three 2-stride convolution layers followed by Instance Normalization and LeakyReLU to make quick judgments, which is inspired by FCN [12]. Due to the flexibility of FCN, we can easily add layers to achieve a progressively growing effect.

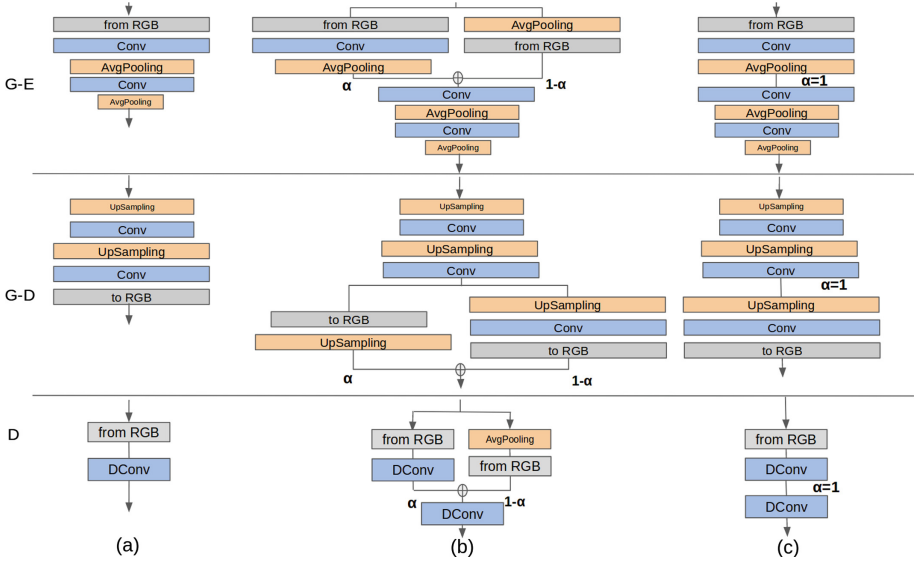


Fig. 3. When growing the layers of the encoder of the generator (G-E), the decoder of the generator (G-D) and discriminator (D), we fade in the new layers smoothly. This example illustrates the translation of the deepening parts, (a) to (c). During the transition (b), we grow α linearly from 0 to 1. fromRGB represents RGB to feature vectors, using the 1×1 convolution layers. toRGB is feature vectors to RGB. Conv means 2 stride-1 3×3 -convolution layers. Dconv is a stride-2 3×3 -convolution layer. When training the discriminator, we feed in real images that are downsampled to increase the judgement on semantic information

3.2 Progressive Growing Strategy

We adopted the progressive growing method of PG-GAN and modified it to suit the encoder-decoder style since the PG-GAN only generates images from random noise of 1×512 codes. The progressively added layers method is shown in Fig. 3. When the network has been trained after adequate training epochs, the progressive growing stage will begin. Firstly, the input image size needs to be enlarged from the original 256×256 to 512×512 . Since we softly add the layer, we use two ways to gradually shade, shown in Fig. 3. For the original round, there will be a new pooling layer before fromRGB to adjust the size because it can only accept 256×256 images. For the other way, the growing layers, which consists of a fromRGB layer of 512×512 and two 1-stride convolution layers with average sampling, will be gradually added to the well-trained structure. There are two weights α and $1 - \alpha$ working on growing layers and original layers separately. With α growing from 1 to 0, the original way will be gradually abandoned and adding layers will progressively integrated well with other parts of this network and a new architecture will be completed. In the decoder part, the process is

similar to that in the encoder. Due to the above idea, our method has great training stability.

3.3 Attention Block

Our attention block aims to find the target domain part and the source part in the pictures. If we add layers into the attention block with the training growing, it will destroy the well-trained attention block, which will need to train again in turn. On the contrary, we keep input 256×256 size images, using bicubic [13] to increase the size to maintain the stability. Like Wang et al’s work [14], we use residual units in our network to increase the accuracy of an attention block.

3.4 Training

The work of domain translation is using a generator G_{st} that translates input image s from a source domain into s' in target domain which is based on a possibility of P_t . At first, we use an attention network A_s , which can locate the source domain part in the images. For the output of the A_s , it is an attention map with per-pixel $[0, 1]$, allowing the network to learn how to compose edges. After the attention block, we can get an image only with the domain part $A_s(s)$ and an image only with the background $1-A_s(s)$, and the other part is just pixels with zero value. Finally, we put the domain part inside the generation and can get the target domain image. We use ‘ \odot ’ to represent the element-wise product. Thus, the mapping from the source domain to the target domain is:

$$s' = (1 - A_s(s)) \odot s + A_s(s) \odot G_{st}(s) \quad (1)$$

We use the progressively growing method to deepen the network, which can handle the large scale images. Before adding a new layer, the model should be fully trained. Through a lot of experiments, we observed that after 100 epochs, it would change a little in the original model. Therefore, after 100 epochs, we will grow the layers in the generation and attention block. We use the G_{stnew} to represent the latest generation and use A_s^* (we do not change the attention block after 30 epochs) as the latest attention block. Using α as weight in progress. The progressively mapping is:

$$s' = (1 - A_s^*(s)) \odot s + (\alpha G_{stnew} + (1 - \alpha)G_{st})(A_s^*(s) \odot s) \quad (2)$$

We use F_{st} and F_{ts} to represent the domain translating. D_t and D_s present the process of discriminators. So the adversarial loss function can be shown as:

$$\mathcal{L}_{adv}^s(F_{st}, A_s, D_s) = \mathbb{E}_{t \sim P_t(t)}[\log D_t(t)] + \mathbb{E}_{s \sim P_s(s)}[\log 1 - D_t(s')] \quad (3)$$

In addition, we enforce network by using cycle consistency loss: calculate the difference between original image s and inverse mapping image s'' , which is s transferred back to original domain by F_{st} and F_{ts} . This process is shown below:

$$\mathcal{L}_{cyc}^s(s, s'') = \|s - s''\|_1 \quad (4)$$

The cycle consistency loss can further reduce the space of possible mapping functions and increase the attention block. Finally, we combine the attention loss and cycle consistency as:

$$\mathcal{L}(F_{st}, F_{ts}, A_s, A_t, D_s, D_t) = \mathcal{L}_{adv}^s + \mathcal{L}_{adv}^t + \lambda(\mathcal{L}_{cyc}^s + \mathcal{L}_{cyc}^t) \quad (5)$$

The optimal parameters of λ e obtained by solving the minimax optimization problem:

$$F_{st}^*, F_{ts}^*, A_s^*, A_t^*, D_s^*, D_t^* = \underset{F_{st}, F_{ts}, A_s, A_t}{\operatorname{argmin}} (\underset{D_s, D_t}{\operatorname{argmax}} \mathcal{L}(F_{st}, F_{ts}, A_s, A_t, D_s, D_t)) \quad (6)$$

For discriminator. At first, the attention block is not precise enough if we just focus on the target part, which will cause the model collapse by combining the information of the background, e.g., in the horse2zebra is the living condition of zebra. To overcome this problem, we train the discriminator with the full image before the first 30 epochs and switch to only the attention part after attention block has developed.

Unpaired image translation generate the pictures will also influenced by the background. Unlike traditional attention block, we should make the boundary sharper to decrease the influence of background. We calculate the attention map as follows:

$$t_{new} = \begin{cases} t & \text{if } A_t(t) > \tau \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

$$s'_{new} = \begin{cases} F_{st}(s) & \text{if } A_s(s) > \tau \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

t_{new} and s'_{new} are masked versions of target sample t and translated source sample s' , which only contain pixels exceeding a user-defined attention threshold τ , which we set to 0.1.

Finally, we update the adversarial loss L of Eq. (3) to:

$$\mathcal{L}_{sdv}^s(F_{st}, A_s, D_t) = \mathbb{E}_{t \sim P_t(t)}[\log D_t(t_{new})] + \mathbb{E}_{s \sim P_s(s)}[\log 1 - D_t(s'_{new})] \quad (9)$$

When optimizing the objective in Eq. (8) beyond 30 epochs, real image inputs to the discriminator is now also dependent on the learned attention maps. This can lead the model to collapse if the training is not performed carefully. For instance, if the mask returned by the attention network is always zero.

$$\mathcal{L}_{sdv}^s(F_{st}, D_t) = \mathbb{E}_{t \sim P_t(t)}[\log D_t(t)] + \mathbb{E}_{s \sim P_s(s)}[\log 1 - D_t(s')] \quad (10)$$

Our model is based on the circulation from source domain to target domain, and back. Which is shown as $\phi_s \rightarrow \phi_{st} \rightarrow \phi_{sts}$, so the cycle consistency is same as function (4). To combine them, the full object is:

$$\mathcal{L}(F_{st}, F_{ts}, D_s, D_t) = \mathcal{L}_{adv}^s + \mathcal{L}_{adv}^t + \lambda(\mathcal{L}_{cyc}^s + \mathcal{L}_{cyc}^t) \quad (11)$$

The solution is similar to the model with attention block, just without the attention part, which is:

$$F^*_{st}, F^*_{ts}, D^*_s, D^*_t = \underset{F^*_{st}, F^*_{ts}, D^*_s, D^*_t}{\operatorname{argmin}}(\operatorname{argmax}\mathcal{L}(F^*_{st}, F^*_{ts}, D^*_s, D^*_t)) \quad (12)$$

4 Experiments

4.1 Training Setting

We use the ‘Apple to Orange’ (A2O) and ‘Horse to Zebra’ (H2Z) datasets provided by Zhu et al. [1] to train our model with attention block since such images have exact foreground object. For our model without attention block, we choose the datasets celeba datasets HD from Karras et al. [5], summer2winter(Yosemite) and monet2photo, which are also from CycleGan [1].

We adopt CycleGAN’s notation [1], “c3s1- k -R” denotes a 3*3 convolution with stride 1 and k filters, followed by a ReLU activation (‘R’), while Leaky ReLU activation with slope 0.2 (‘LR’). “ap” denotes an average pooling layer halving the input layer. “rk” denotes a residual block formed by two 3*3 convolutions with k filters, stride 1 and a ReLU activation. “up” denotes a upsampling layer doubling the heights and widths of its input. A Sigmoid activation is indicated by ‘S’ and ‘tanh’ by ‘T’. We apply Instance Normalization after all layers apart from the last layer.

Final generator architecture is: c1s1-32-R, c3s1-64-R, c3s1-64-R, ap, c3s1-64-R, c3s1-64-R, ap, c3s1-128-R, c3s1-128-R, ap, r128, r128, r128, r128, r128, r128, r128, r128, up, c3s1-64-LR, c3s1-64-LR, up, c3s1-32-LR, c3s1-32-LR, up, c3s1-32-LR, c3s1-32-LR, c1s1-3-T.

Attention block architecture is: c7s1-32-R, c3s2-64-R, r64, up, c3s1-64-R, up, c3s1-32-R, c7s1-1-S.

Final discriminator architecture is: c3s1-64-LR, c4s2-32-LR, c4s2-64-LR, c4s2-128-LR, c4s2-256-LR, c4s1-512-LR, c4s1-1.

Similar to CycleGAN, we use the Adam solver with a batch size of 1. All networks were trained from scratch with a learning rate of 0.0002. We keep the same learning rate for the 200 epochs. Weights are initialized from a Gaussian distribution $\mathcal{N}(0,0.02)$. Layers are added in 140, 170 epochs.

4.2 PG-Method and Attention Block

Observing the Fig. 5, we can see that the generated images are getting more and more fine details through training, which means our progressively growing method work. When it in step1, there are only limited strips on the generated zebras, but as the step grows, the strips are getting more and more. Finally,



Fig. 4. Translation results. From top to bottom. Zebra to horse, horse to zebra, apple to orange, summer to winter, winter to summer. For the first three translations, our results are generated with attention block, and for the last two translation, attention block is not used. (Color figure online)

all the generated zebras are covered with strips, which makes them really like zebras.

The function of the attention block is to tract the domain part in the image, which will protect the background information while in translation. In Fig. 6, looking at the attention maps (the grey images), each of them can accurately find the source domain. As a result, shown in the photos after the attention block, the generated pictures will have the same background as the original images have.

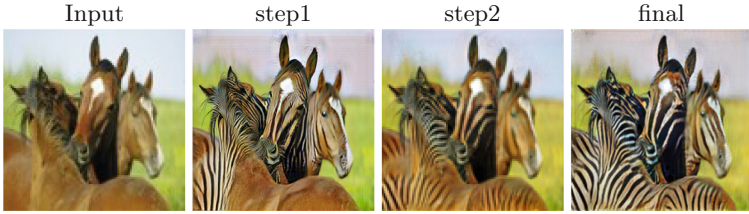


Fig. 5. Domain translation for generating a zebra images by a horse image. Results of step1 (fully trained with 256×256 size images, which is same as CycleGAN), step2 (fully trained with 512×512 images), and step3 (fully trained with 1024×1024 pictures, the model already finished growing). With the layers growing, some fine details are added. The strips of generated zebra is adding with the step increasing.



Fig. 6. Results of attention block for zebra to horse and horse to zebra domain translation of four group. The order inside each group is input image, generated image without attention map, attention map and generated image with attention map. The attention block can correctly tract the domain part inside the images.

4.3 Baselines

Nowadays, there are many famous GANs performing well in domain transferring using different loss. CycleGAN [1] with least-squared GAN [16] loss and DiscoGAN [15] with Standard loss [17] use a circulation to train adversarial network. Dual GAN [19] uses Wasserstein GAN loss [18] to solve the model collapse. To prove our model really work well on high pixels images, we compare our model with CycleGAN [1], CycleGAN [1] with progressively growing method [5] and DiscoGAN [15] on 1024×1024 images.

4.4 Qualitative Results and Quantitative Results

Figure 4 shows the results of Horse2zebra, Apple2Orange, Summer2Winter, Monet2Photo, and blond2brown datasets. Although CycleGAN has a strong ability in domain transfer, the background will be changed by trained mapping function due to loss function, which is based on whole the image. Moreover, when we zoom in the image from CycleGAN, obvious checkerboard artifact resulted from transposed convolution layers can be observed. The simple combination of PG-GAN and CycleGAN do not have good performance. Because the receptive field changes a lot when new layers are added, the model collapsed is easier

Table 1. VGG perception for each model

Model	H2Z	Z2H	A2O	O2A	M2P	P2M
VGG(accuracy)	0.99	0.99	0.96	0.96	0.98	0.98
Ours	0.78	0.94	0.84	0.83	0.84	0.27
CycleGAN	0.82	0.90	0.77	0.87	0.83	0.11
PG CycleGAN	0.75	0.65	0.62	0.59	0.72	0.09
Disco GAN	0.63	0.19	0.80	0.19	0.74	0.35

^a Higher scores mean better model.

^b H (horse), Z(zebra), A(apple), O(orange), M(monet), P(photo)

to occur. DiscoGAN focuses on the relationship between two domains, but can only realize unidirectional domain translation, such as horse2zebra in Fig. 4. By incorporating the progressive growing strategy, attention block and replacing transpose convolution with bilinear interpolation upsampling, our results have less checkerboard effect, more natural background and stronger ability of domain transfer. Our model successfully makes output more realistic compared with other models and manages to solve the checkerboard artifacts.

We use the VGG-16 network [19] to quantitatively evaluate the authenticity of our generated images. VGG-16 is a classical model in Image Identification. Comparing with AlexNet [20], VGG-16 used stacked small convolution kernels increasing the depth of the network with fewer parameters. We prepare a unique VGG-16 network for each datasets, expect winter2summer that VGG-16 only has 70% accuracy. For the training datasets are the same with the datasets we used in domain translation. As we wanted our domain translation is more natural, which means our results should have a higher grade in VGG-16 network. Therefore, we calculate the accuracy of each dataset in Table 1, where we also list the accuracy of VGG-16 network for test images.

Our approach reaches the highest score in most domain translation, while the CycleGAN is the second, which means it does well in domain translating. Although PG CycleGAN uses the progressive growing method, it is third one of all, because the model is not stable enough, then adding layer will always just learn to change the color instead of the domain translation. Because of the loss function used by DiscoGAN, it can reach good results in one direction domain translation. Finally, comparing with these GANs, it is obvious that our model deepens the understanding of the semantic information through progressive training and enhances the vision of the attention block (Figs. 7 and 8).

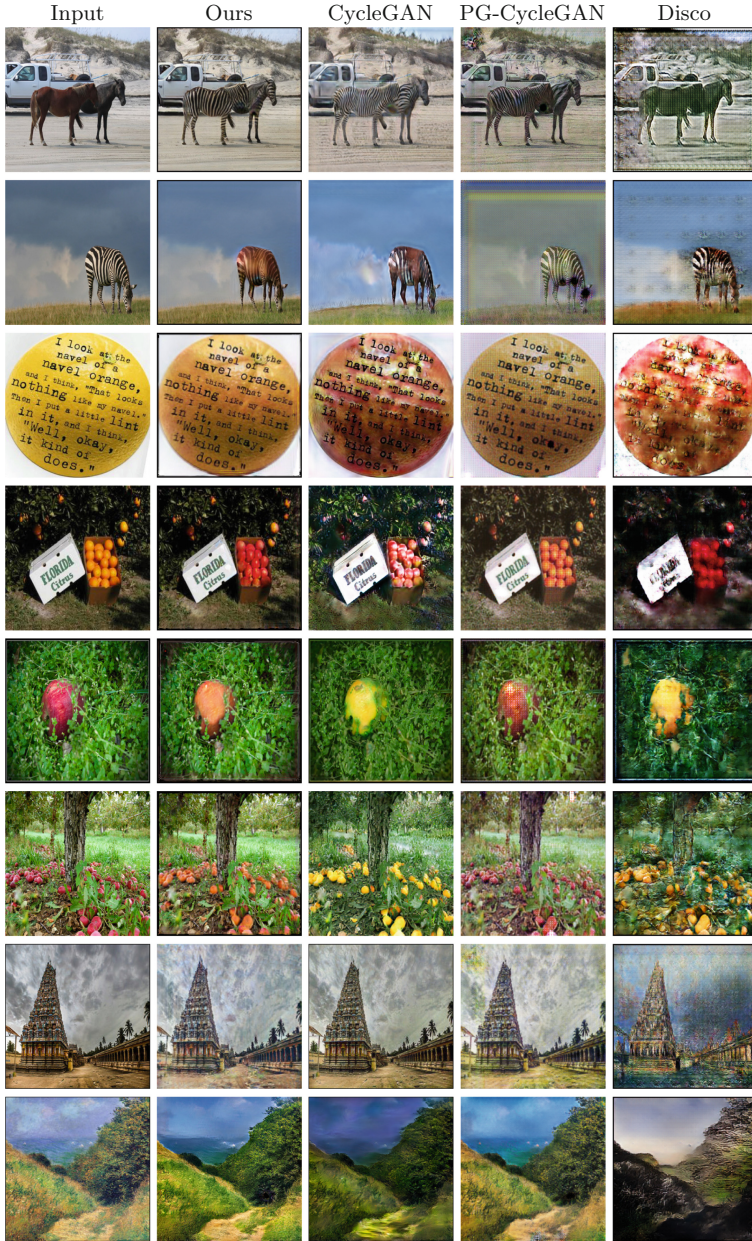


Fig. 7. Translation results. From top to bottom. Horse to zebra, zebra to horse, orange to apple, apple to orange, picture to monet, monet to picture. For the first five translations, our results are generated with attention block, and for the last three translation, attention block is not used. (Color figure online)

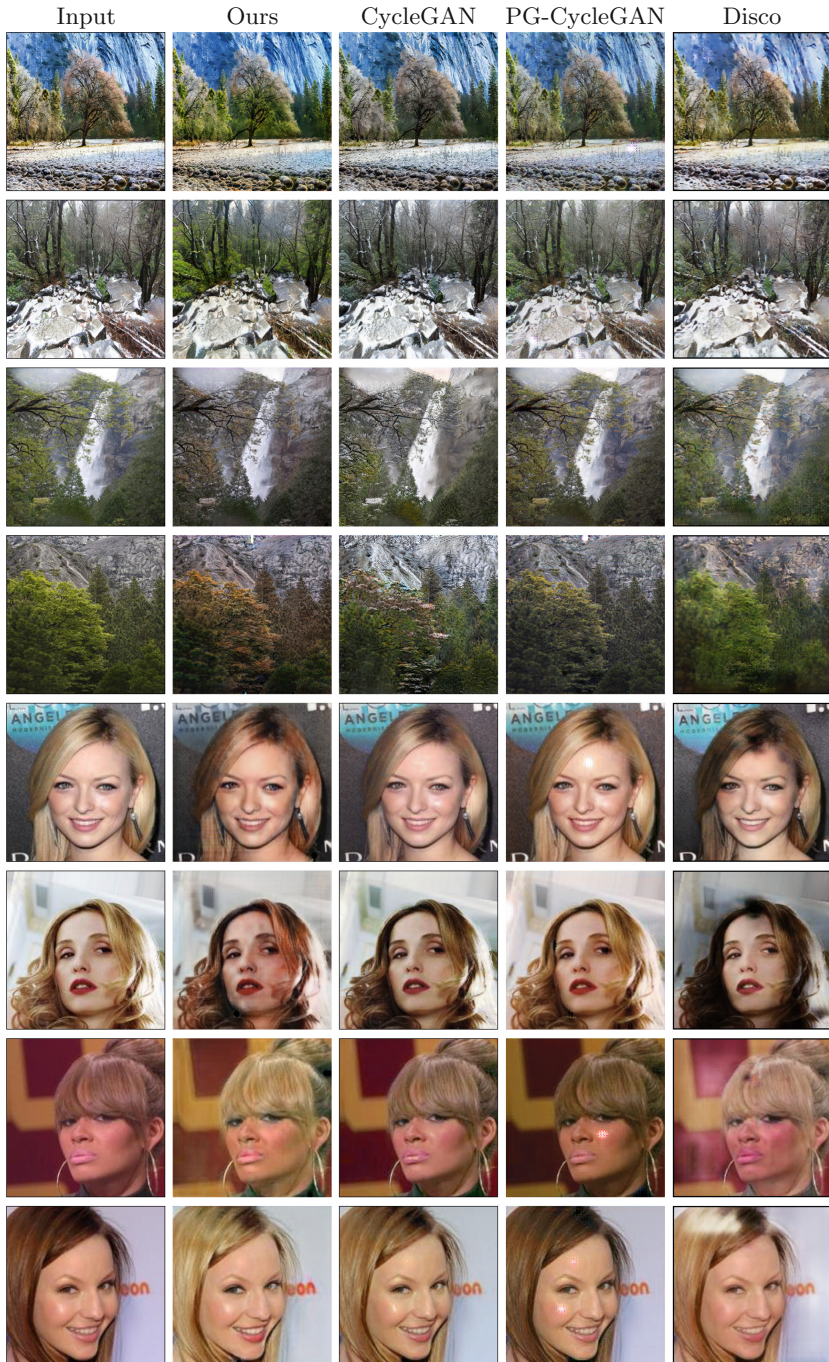


Fig. 8. Translation results. From top to bottom. Winter to summer, summer to winter, blond hair to brown hair, brown hair to blond hair. Attention Block is not used.

5 Conclusion

Simply combination of a progressively growing method with CycleGAN will easily cause model collapse. In this paper, we present a more stable GAN-PG-Att-CycleGAN. The architecture trains an adversarial network gradually with the help of attention block, and fix the generator to reach the goal. Our method can greatly reduce the damage of the deep layer to the spatial information. Besides, with the help of the increased number of layers and skip connection, we can generate images with more natural textures.

References

1. Zhu, J.-Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: ICCV (2017)
2. Pecina, P., et al.: Adaptation of machine translation for multilingual information retrieval in the medical domain. *Artif. Intell. Med.* **61**(3), 165–185 (2014)
3. Wei, L., Zhang, S., Gao, W., Tian, Q.: Person transfer GAN to bridge domain gap for person re-identification. In: CVPR (2018)
4. Bansal, A., Ma, S., Ramanan, D., Sheikh, Y.: Recycle-GAN: unsupervised video retargeting. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11209, pp. 122–138. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01228-1_8
5. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of GANs for improved quality, stability and variation. In: ICLR (2018)
6. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
7. Mejjati, Y.A., Richardt, C., Tompkin, J., Cosker, D.: Unsupervised attention-guided image-to-image translation. In: NIPS (2018)
8. Odena, A., Dumoulin, V., Olah, C.: Deconvolution and checkerboard artifacts. *Distill* **1**, e3 (2016)
9. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual Losses for Real-Time Style Transfer and Super-Resolution. arXiv preprint [arXiv:1603.08155](https://arxiv.org/abs/1603.08155) (2016)
10. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Instance normalization: the missing ingredient for fast stylization. arXiv preprint [arXiv:1607.08022](https://arxiv.org/abs/1607.08022) (2016)
11. Zhang, H., et al.: StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks. arXiv preprint [arXiv:1612.03242](https://arxiv.org/abs/1612.03242) (2016)
12. Long, J., Shelhamer, E.: Fully convolutional networks for semantic segmentation. In: CVPR (2015)
13. Bellman, R., Kashef, B., Vasudevan, R.: Dynamic programming and bicubic spline interpolation. *J. Math. Anal. Appl.* **44**, 160–174 (1973)
14. Wang, F., et al.: Residual Attention Network for Image Classification. arXiv preprint [arXiv:1704.06904](https://arxiv.org/abs/1704.06904) (2017)
15. Kim, T., Cha, M., Kim, H., Lee, J.K., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. In: International Conference on Machine Learning (2017)
16. Mao, X., et al.: Least Squares Generative Adversarial Networks. arXiv preprint [arXiv:1611.04076](https://arxiv.org/abs/1611.04076) (2016)

17. Goodfellow, I., et al.: Generative adversarial nets. In: NIPS (2014)
18. Arjovsky, M., Chinatala, S., Bottou, L.: Wassertein GAN. arXiv preprint [arXiv:1701.07875](https://arxiv.org/abs/1701.07875) (2017)
19. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556) (2016)
20. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: NIPS (2012)



Human Motion Generation Based on GAN Toward Unsupervised 3D Human Pose Estimation

Sasuke Yamane, Hirotake Yamazoe^(✉), and Joo-Ho Lee

Ritsumeikan University, Kusatsu, Shiga 525-8577, Japan
yamazoe@fc.ritsumei.ac.jp

Abstract. In this paper, we propose a method for generating joint angle sequences toward unsupervised 3D human pose estimation. Many researchers have proposed human pose estimation methods. So far, however, most methods have problems that require a large amount of images with supervised pose datasets to learn pose estimation models. Building such datasets is a time-consuming task. Thus, we aim to propose a method that can estimate 3D human poses without requiring training data with known poses. Toward this goal, we propose a GAN-based method for human motion generation and an optimization-based human pose estimation method. The proposed method consists of a generator that generates human pose sequence, a renderer that renders human images by changing 3D meshes based on the pose sequences generated, and a discriminator that discriminates between generated images and training data. Through an experiment based on simulated walking images, we confirmed that the proposed method can estimate the poses of body parts that are not occluded.

Keywords: 3D human pose estimation · Unsupervised learning · Generative adversarial networks

1 Introduction

Human pose estimation from image sequences has various applications such as activity recognition, user interface, and others. Thus, many researchers have proposed human pose estimation methods [4, 10]. In recent years, many deep learning-based methods for estimating human poses have been proposed. By using these methods, we can obtain accurate human poses from RGB images.

3D human pose estimation methods are mainly divided into two categories. One is a two-step method that first estimates the positions of joints in the 2D image and then estimates 3D human poses based on the 2D joint positions [6, 7]. The problem with this type of method is that the final 3D pose estimation accuracy depends on the estimation accuracy of the 2D joint positions. In addition, since the 3D pose estimations are based on 2D joint positions, such methods may not be able to utilize image features from the entire input image.

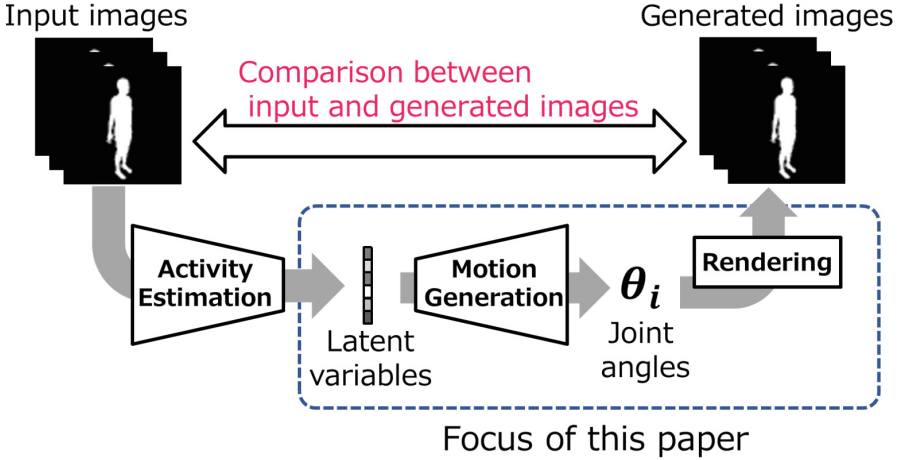


Fig. 1. Final goal of this research

Another type of method involves estimating the 3D human poses directly from the image [1, 3, 12, 13, 15]. In these methods, 3D pose estimation can be achieved directly from RGB images by learning from supervised datasets with known poses as training data. These methods require a large amount of supervised pose datasets to learn human pose estimation models. Building such datasets is a time-consuming task, and preparing a large amount of such data is difficult.

In contrast, in this research, we aim to achieve a method that can estimate 3D human poses without requiring training data of known poses. Figure 1 shows the final framework that this research aims to realize. First, the latent vectors that represent actions or behaviors are obtained from the input image sequences. Next, human joint angle sequences are generated from these latent vectors. Finally, human images are rendered from the joint angle sequences. Here, if we can learn to generate an image sequence that matches the input image sequences through a series of these processes, the joint angles of the input images can be estimated as intermediate results (joint angles in Fig. 1).

In this paper, we focus on the latter part of the Fig. 1 (inside the dotted line frame)—that is, the processes from the generation of the joint angle sequences to the generation of the human image sequences using latent vectors as input. We aim to realize these processes by using a GAN-based framework. Figure 2 shows the flow of the proposed method. Here, instead of latent vectors, we employ noise vectors. The proposed method consists of a motion generation model that generates joint angle sequences, and a renderer that renders human image sequences by deforming a 3D mesh model based on the generated joint angles. These generative models are trained so that the differences between the generated image sequences and the training data are minimized. In addition, we cannot estimate human poses by using only the latter part of Fig. 1, we also propose a joint angle

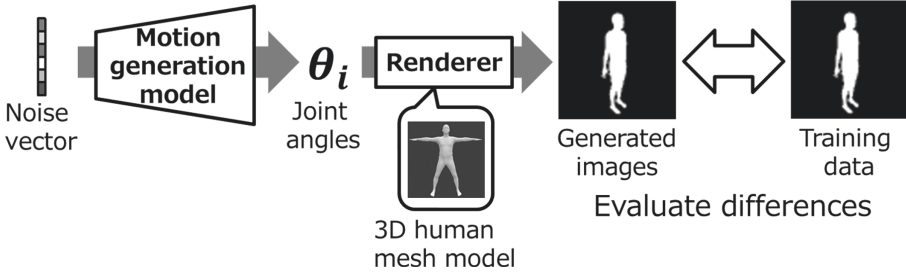


Fig. 2. Overview of proposed method

sequence estimation method for this purpose. In this paper, we focus on walking motions and show the effectiveness of the proposed method.

The contributions of this paper are as follows:

- a new scheme toward unsupervised 3D human pose estimation
- a human motion generation model that can render human images that match training datasets
- a network structure for this purpose combining differentiable modules

2 Proposed Method

2.1 Overview

An overview of the proposed method is shown in Fig. 2. The proposed method consists of a motion generation model that generates joint angle sequences θ_i , and a renderer that renders human image sequences by deforming a 3D mesh model based on the generated joint angles. The generative model is learned to reduce the differences between the training data and the generated images. After training, we can generate joint angle sequences that match the training dataset images.

For the motion generation model that generates joint angle sequences, we employ long short-term memory (LSTM). The skinned multi-person linear (SMPL) model [5] is used in the process that deforms the 3D human mesh based on the joint angle sequences, and we render human images from the 3D mesh by using the neural renderer model [2]. These processes are constructed with differentiable models, and errors in the rendered images can be backpropagated to the motion generation model. Thus, the motion generation model can be learned from errors in the rendered image sequences. The proposed model is designed based on a GAN framework and can be learned in the end-to-end manner.

2.2 Detailed Configuration

Figure 3 shows the detailed configuration of the proposed method. It consists of a motion generator that generates joint angle sequences, a discriminator that

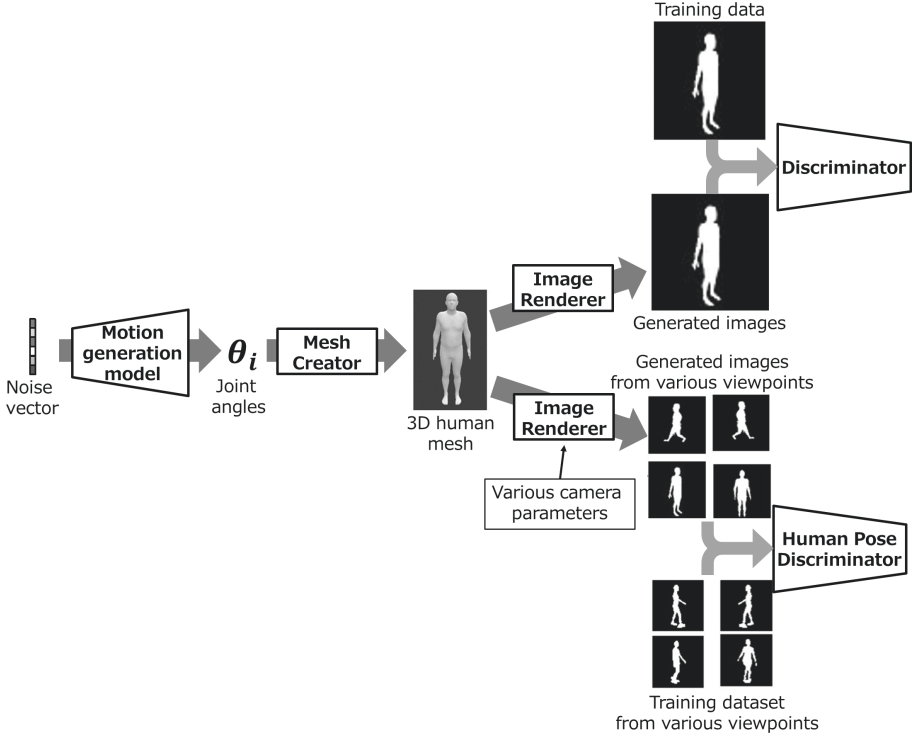


Fig. 3. Configuration of the proposed method

discriminates between the generated images and training data, a human pose discriminator for learning human body structures and their limitations, and a renderer. The renderer consists of a mesh creator that transforms a 3D human body mesh based on the generated joint angles and an image renderer that renders the human body mesh into 2D images.

Generator. The generator is implemented as a two-layer LSTM. The input of the generator is the noise vector $\mathbf{z} \in \mathbb{R}^{64}$ and the output is the joint angle sequences $\boldsymbol{\theta} \in \mathbb{R}^{l \times 3 \times K}$. l is the number of frames of one image sequence and K is the number of joints. In the current implementation, $l = 8$ and $K = 8$.

Discriminator. The discriminator was designed by referencing the structure of the discriminator in DCGAN [9] and implemented using a 3D convolutional neural network (3D Conv) [14] that can handle image sequences. Our discriminator consists of four 3D Conv layers, four max pooling layers, and one fully connected layer, and it outputs the probability of whether the input image sequences are rendered images or training data. We use leaky ReLUs for activation functions. Table 1 shows the detailed configuration of the discriminator.

Table 1. Configuration of discriminator

Type	Kernel	Stride	Outputs
3D Conv	3×3	2×2	16
Max pooling	3×3	2×2	16
3D Conv	3×3	2×2	32
Max pooling	3×3	2×2	32
3D Conv	3×3	2×2	64
Max pooling	3×3	2×2	64
3D Conv	1×1	1×1	64
Max pooling	1×1	1×1	64
FC	–	–	1

Mesh Creator. We used the skinned multi-person linear (SMPL) model [5] as a mesh creator. This model can generate differentiable 3D meshes $M(\boldsymbol{\theta}) \in \mathbb{R}^{3*N}$ when 3D joint angles $\boldsymbol{\theta}_i$ are inputted. N is the number of mesh vertices and $N = 6980$. The 3D joint angles $\boldsymbol{\theta}_i \in \mathbb{R}^{3*K}$ are the axis-angle representations. The number of the joints is $K = 23$. The body parameter $\boldsymbol{\beta} \in \mathbb{R}^{10}$ is a 10-dimensional vector representing the body shape. These parameters in the mesh creator do not require training and therefore do not need to change during the training phase. However, in the current implementation, the number of joints is reduced and only the hip, knee, shoulder and elbow joints ($K = 8$) are focused on. In addition, we employ the shape parameters of the template body mesh as the body shape parameters.

Image Renderer. The neural renderer from [2] is used as the image renderer. This is a differentiable image rendering function that outputs human images $\hat{x} \in \mathbb{R}^{l*h*w}$ based on the 3D human mesh $M(\boldsymbol{\theta}) \in \mathbb{R}^{l*3*N}$ outputted by the mesh creator and the camera parameters $\mathbf{c} \in \mathbb{R}^3$. Here, h and w are the height and width of the input image, and the camera parameters are expressed as $\mathbf{c} = [distance, elevation, angle]$. Since the parameters in the image renderer do not require training, they do not need to change during the training phase. In addition, in the following experiment, the camera parameters are already known.

Human Pose Discriminator. The four modules described above are the basic parts of the proposed method. However, with only these modules, the 3D joint angles generated by the generator and the discriminate accuracy of the discriminator were dependent on the viewpoint of the training data. Furthermore, since the generator did not contain constraints of human body structures, the generator may generate joint angles that humans cannot pose at. Thus, in our method, we introduce a human pose discriminator that learns the constraints of the human body structure.

Table 2. Configuration of human pose discriminator

Type	Kernel	Stride	Outputs
3D Conv	3×3	2×2	16
Max pooling	3×3	2×2	16
3D Conv	3×3	2×2	32
Max pooling	3×3	2×2	32
3D Conv	3×3	2×2	64
Max pooling	3×3	2×2	64
3D Conv	1×1	1×1	64
Max pooling	1×1	1×1	64
FC	–	–	1

The human pose discriminator consists of four convolution layers, four max pooling layers, and one fully connected layer, as shown in Table 2, and discriminates whether the input images were training data or generated by the generator. For training the human pose discriminator, we employed human images captured from various viewpoints. The input images were taken from various viewpoints rendered by the image renderer in which the camera parameters were changed at random.

The objective functions of the generator, the discriminator, and the human pose discriminator are expressed by the following equations.

$$\begin{aligned} \min L(G) = & \mathbb{E}_{p(z)}[\log(1 - D(\hat{x}; G(z)))] \\ & + \alpha(\mathbb{E}_{p(z)}[\log(1 - Reg(\hat{x}; G(z)))] \end{aligned} \quad (1)$$

$$\begin{aligned} \max L(D) = & \mathbb{E}_{pdata(x)}[\log D(x)] \\ & + \mathbb{E}_{p(z)}[\log(1 - D(\hat{x}; G(z)))] \end{aligned} \quad (2)$$

$$\begin{aligned} \max L(Reg) = & \mathbb{E}_{pdata(x)}[\log Reg(x)] \\ & + \mathbb{E}_{p(z)}[\log(1 - Reg(\hat{x}; G(z)))] \end{aligned} \quad (3)$$

where $G()$ is the generator, $D()$ is the discriminator, and $Reg()$ is the human pose discriminator, z is the noise vector, x is the training data, \hat{x} is the rendered images, and α is the weight of the regularization term using the human pose discriminator.

2.3 Joint Angle Estimation

As described above, this paper focused on only the latter part of Fig. 1, we cannot estimate 3D poses of input images with only these modules. Thus, we implemented a method for estimating the joint angles from newly inputted test images. Figure 4 shows the process flow of the method.

First, the trained models generate the joint angle sequences and render the human image sequences. Then, by minimizing the errors between the rendered

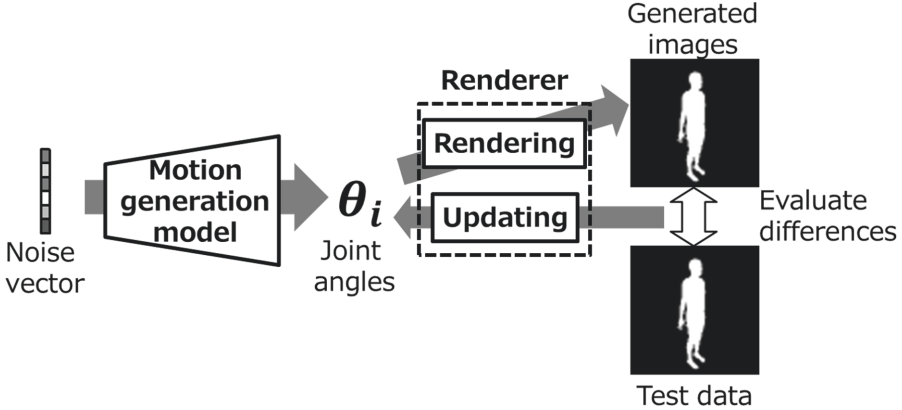


Fig. 4. Estimation method for estimating joint angles of input images

images and the inputted images, we can find accurate angles of joints in the inputted images. The objective function for estimating the joint angles of the inputted images are as follows:

$$\min Loss = \frac{1}{n} \sum_{n=1} ||x_n - \hat{x}_n||^2 \quad (4)$$

where N is the number of the images, x is the inputted images, and \hat{x} is the rendered images. In the current implementation, we assume the camera parameters are known, but we can consider these parameters as unknown and estimate them.

3 Experiment

To evaluate the effectiveness of the proposed method, we conducted the following experiment. For training data, we generated CG data by using the SMPL model and the neural renderer. Here, we focused on human walking. We generated CG images of human walking scenes based on [8], and randomly changed the ranges of each angle magnitude in order to express various gait patterns. To train the human pose discriminator, we employed the OU-ISIR Gait Database, Multi-View Large Population Dataset (OU-MVLP) [11], which includes images of walking scenes of various people from multiple viewpoints.

In the experiment, we generated and employed 50 videos of two viewpoints as the training data. These image sequences consist of eight frames. In the training phase, we adopted the model at the epoch that obtained the minimum mean square error between the training data and the images rendered by the proposed model as the final trained model. The learning rate was 0.001, the batch size was 5, and the weight of the regularization term α was 0.01. Adam was adopted as the optimization method. The two camera viewpoints in the training data were

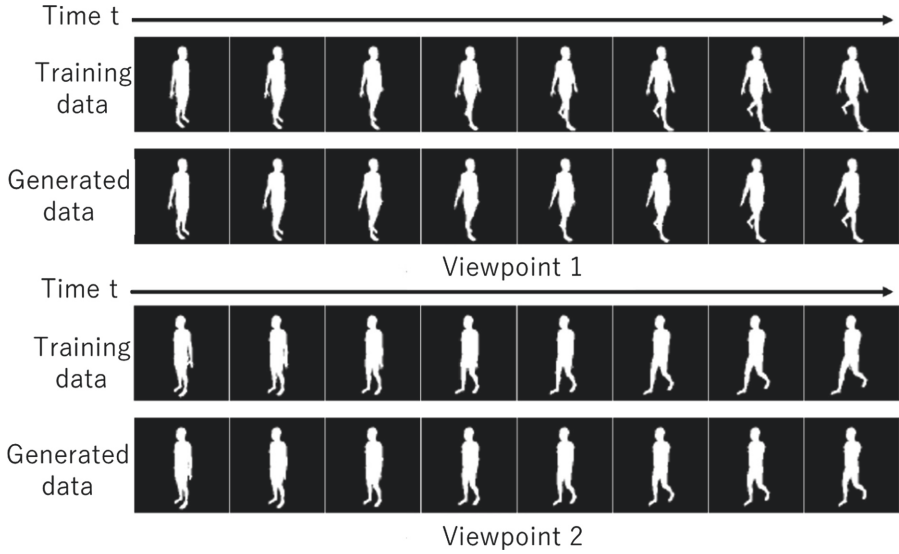


Fig. 5. Example of generated images from trained model



Fig. 6. Image generation results of proposed model from test images

alternated for each batch. The generated images and the training data were alternately inputted in each batch to the discriminator. Training took 10,8264 seconds (about 30 h).

Examples of the generated images are shown in Fig. 5. As can be seen, the proposed model can generate images of similar appearance to the training data.

Next, we estimated the joint angles from the test data. Figures 6 and 7 show these results. The results showed that the hip and knee joint angles can be estimated correctly. On the other hand, the shoulder and elbow joint angles cannot be estimated correctly. The reason for such different tendencies seems to be that the joint angles are estimated based on the mean square error of the silhouette. Therefore, in scenes in which the arms and torso overlap, the arm movements are difficult to estimate accurately. In addition, in the training phase, since we used images from two viewpoints as training data, we can obtain the arm movements even if silhouette images are employed. However, in the

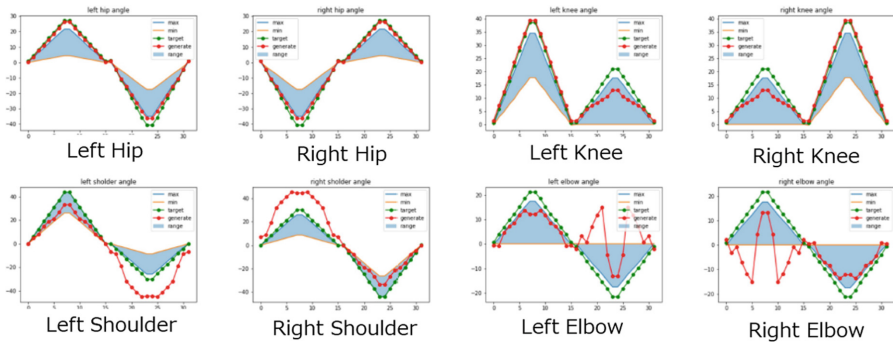


Fig. 7. Joint angle estimation results

estimation phase, we employed only one viewpoint; thus, we consider that only silhouette images were insufficient for estimating arm movements.

4 Conclusion

In this paper, we proposed a 3D human pose estimation method that did not require training data with known poses. Experiments using CG data of simulated walking confirmed that the proposed method can estimate the poses of the body parts which were not occluded.

Future works can include investigations of an optimization method using RGB images instead of silhouette images and a method that can estimate the poses of body parts even when they will be occluded. Implementation of the whole method shown in Fig. 1 is also a matter for future works.

Acknowledgement. This work was supported by JSPS KAKENHI Grant Number JP17K00372 and JP18K11383.

References

1. Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7122–7131 (2018)
2. Kato, H., Ushiku, Y., Harada, T.: Neural 3D mesh renderer. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3907–3916 (2018)
3. Kulkarni, T.D., Kohli, P., Tenenbaum, J.B., Mansinghka, V.: Picture: a probabilistic programming language for scene perception. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4390–4399 (2015)
4. Liu, Z., Zhu, J., Bu, J., Chen, C.: A survey of human pose estimation: the body parts parsing based methods. *J. Vis. Commun. Image Represent.* **32**, 10–19 (2015)
5. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: SMPL: a skinned multi-person linear model. *ACM Trans. Graph. (TOG)* **34**(6), 248 (2015)

6. Martinez, J., Hossain, R., Romero, J., Little, J.J.: A simple yet effective baseline for 3D human pose estimation. In: International Conference on Computer Vision, vol. 1, p. 5 (2017)
7. Moreno-Noguer, F.: 3D human pose estimation from a single image via distance matrix regression. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1561–1570. IEEE (2017)
8. Murray, M.P., Kory, R.C., Clarkson, B.H., Sepic, S.: Comparison of free and fast speed walking patterns of normal men. *Am. J. Phys. Med. Rehabil.* **45**(1), 8–24 (1966)
9. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks (2015). arXiv preprint [arXiv:1511.06434](https://arxiv.org/abs/1511.06434)
10. Sarafianos, N., Boteanu, B., Ionescu, B., Kakadiaris, I.A.: 3D human pose estimation: a review of the literature and analysis of covariates. *Comput. Vis. Image Understand.* **152**, 1–20 (2016)
11. Takemura, N., Makihara, Y., Muramatsu, D., Echigo, T., Yagi, Y.: Multi-view large population gait dataset and its performance evaluation for cross-view gait recognition. *IPSJ Trans. Comput. Vis. Appl.* **10**(4), 1–14 (2018)
12. Tan, J., Budvytis, I., Cipolla, R.: Indirect deep structured learning for 3D human body shape and pose prediction. In: Proceedings of the BMVC, London, UK, pp. 4–7 (2017)
13. Toshev, A., Szegedy, C.: Deeppose: human pose estimation via deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1653–1660 (2014)
14. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3D convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4489–4497 (2015)
15. Tung, H.Y., Tung, H.W., Yumer, E., Fragkiadaki, K.: Self-supervised learning of motion capture. In: Advances in Neural Information Processing Systems, pp. 5236–5246 (2017)



vi-MoCoGAN: A Variant of MoCoGAN for Video Generation of Human Hand Gestures Under Different Viewpoints

Thanh-Hai Tran¹(✉) , Viet-Dung Bach¹ , and Huong-Giang Doan² 

¹ International Research Institute MICA,
Hanoi University of Science and Technology, Hanoi, Vietnam
thanh-hai.tran@mica.edu.vn, dungbachviet@gmail.com

² Electric Power University, Hanoi, Vietnam
giangdth@epu.edu.vn

Abstract. This paper presents a method for video generation under different viewpoints. The method gets inspired by MoCoGAN's idea which modelled a video clip in two latent sub-spaces (content and motion) and achieved impressive results recently. However, MoCoGAN and most of existing methods of video generation did not take viewpoint into account so they cannot generate videos from a certain viewpoint, which is a need for data augmentation and advertisement applications. To this end, we propose to follow the idea of conditional GAN and introduce a new variable to control the generated video's view. In addition, to keep the subject consistent during action implementation, we utilize an additional sub-network to generate the content control vector instead of using a random vector. Besides, the objective function for training the network will be modified to measure the similarity of content, action and view of the generated video with the truth one. Preliminary experiments are conducted for generating video clips of dynamic human hand gestures, showing the potential to generate videos under different viewpoints in the future.

Keywords: Data augmentation · Video GAN · Dynamic hand gesture · Multi-view

1 Introduction

Dynamic hand gestures have been shown to be very effective for human machine interaction [5]. Despite the fact that there exist a number of methods for dynamic

This material is based upon work supported by the Air Force Office of Scientific Research under award number FA2386-17-1-4056.

Electronic supplementary material The online version of this chapter (https://doi.org/10.1007/978-981-15-3651-9_11) contains supplementary material, which is available to authorized users.

hand gesture recognition from video, deployment of such methods in practical applications must face many challenges such as low hand resolution; phase variation; complex background and viewpoint changes, among which viewpoint change is the most critical issue [10]. To deal with viewpoint change, ones have to prepare data of dynamic hand gestures observed under different viewpoints for training recognition models. Currently, deep models have been shown to be very

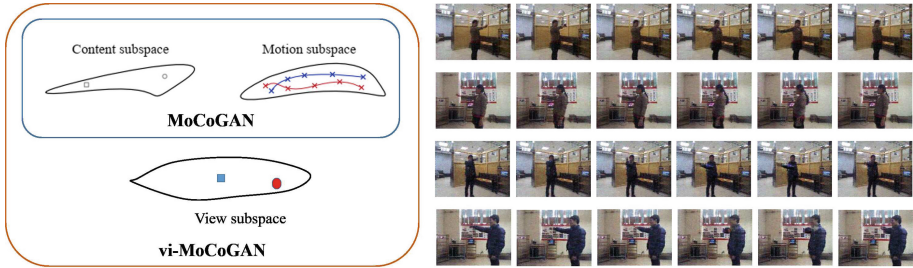


Fig. 1. Video space decomposition: in Motion-Content sub-spaces in MoCoGAN [11] and in Motion-Content-View sub-spaces in our proposed vi-MoCoGAN. The right image sequences (row by row) illustrate key-frames of videos generated from 2 views with 2 subjects (content) for one motion (action).

powerful in many computer vision tasks. However, training deep models always requires big data. Unfortunately, there exist rarely multi-view dynamic hand gestures datasets. In addition, setup for collecting hand gestures by a large number of cameras and subjects is not easy. In a survey given in [6], only one among fifty presented datasets, was collected by multiple cameras, but this dataset concerns sign language, not for human machine interaction. In [4] and [10], a multi-view dataset has been presented but it showed that performance of cross-view recognition is still very limited due to the lack of training data. Some recent works have shown that data augmentation could improve significantly recognition performance. The problem of video generation at a certain viewpoint was not raised in the literature. Until now, this idea for multi-view data generation was only studied for still images (e.g. faces) in CRGAN [9]. A direct application of this approach for video generation is to use CRGAN for generating each separated frame. However, this approach will be very time consuming because it must process frame by frame. In addition, this approach focuses on frame generation, it does not take motion of the object into account.

This paper presents a method for automatic generation of dynamic human hand gestures from different viewpoints which helps to augment data for training deep models of recognition. The ongoing work enriches the real dataset with artificial videos generated from pre-defined viewpoints. Generating videos from a novel viewpoint will be a future work. Our proposed method gets inspired by the idea of MoCoGAN (Motion Content GAN) [11] that modelled a video in two

sub-spaces (motion and content). However, the original MoCoGAN and most of existing methods for video generation did not take viewpoint into account. In our work, we introduce a view control vector into MoCoGAN to learn viewpoints. In addition, we also put another conditional vector generated from an image encoder to ensure that the subject (content) does not change during performing hand gestures. With both additional inputs, the objective function of the whole network will be modified to measure the similarity of content, view and action at the same time. We name our proposed method as **vi-MoCoGAN** with the prefix *vi* standing for viewpoint to distinguish with the original MoCoGAN. Compared to the original MoCoGAN, which decomposes a video clip into motion and content, vi-MoCoGAN decomposes a video clip into three components: motion, content and viewpoint (Fig. 1).

To evaluate the proposed method, we collect a dataset of twelve dynamic hand gestures observed under five viewpoints. This dataset will be used to train vi-MoCoGAN then vi-MoCoGAN generates new samples of dynamic gestures among twelve pre-defined gestures from five views. The experiments show promising results of video generation at different viewpoints. In summary, the contributions of our work are: (i) adapt a video generation network to deal with different viewpoints; (ii) put a constraint on the subject to make it more consistent during gesture/action implementation; (iii) preliminary evaluation of generating dynamic human hand gestures.

2 Related Work

2.1 Existing Works of Video Generation

Video generation is a new topic in computer vision. It opens many applications for example entertainment/advertisement or data augmentation for machine learning algorithms. However, the problem of video generation remains still a big challenge because video is a spatial-temporal sequence which contains objects performing actions. As a consequence, a generative model needs to learn both the appearance and physical motion of the object at the same time. Generative Adversarial Network (GAN) has been widely applied for image generation [2]. Recently, some works inspired the idea of GAN for video generation have been attempted such as VGAN [13], TGAN [7], MCNET [12] and MoCoGAN [11]. MoCoGAN has outperformed TGAN and VGAN on several benchmark datasets such as MUG Facial Expression; Tai-Chi; UCF101. This motivates us to study MoCoGAN and extend it for video generation under different viewpoints.

2.2 Summary of MoCoGAN

The main idea of MoCoGAN is to consider each video as a combination of motion and content. Therefore, the latent space of video clips should be decomposed into two latent sub-spaces which are motion subspace and content subspace. This decomposition facilitates the control of motion and content generation which is absent in existing video generation methods. We assume the latent space of

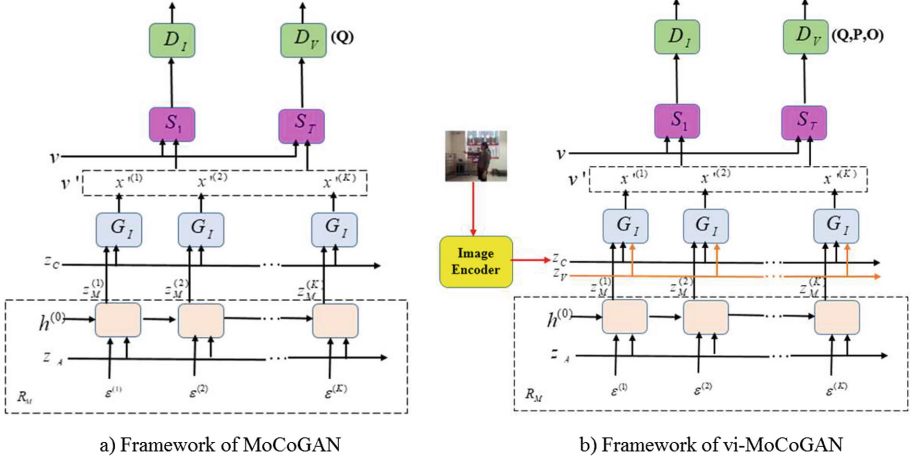


Fig. 2. Frameworks of (a) the original MoCoGAN [11] and (b) the proposed vi-MoCoGAN.

images $Z_I \equiv R^d$. Each $z \in Z_I$ is an image in the space Z_I and a video is a path of length K in the image space $[z^{(1)}, z^{(2)}, \dots, z^{(K)}]$. In MoCoGAN, the latent space Z_I is decomposed into two sub-spaces: the content space $Z_C = R^{d_C}$ and the motion space $Z_M = R^{d_M}$: $Z_I = Z_C \times Z_M$ with $d = d_C + d_M$. The content sub-space is modelled by a Gaussian distribution $z_C \sim p_{Z_C} \equiv \mathcal{N}(z|0, I_{d_C})$ where I_{d_C} is the identity matrix of size $d_C \times d_C$. In a short video, the content remains the same so the same realization of z_C will be used for generating different frames in a video clip. The motion in the video clip is modeled by a path in the motion subspace Z_M . The sequence of vectors to generate a video is represented by Eq. (1):

$$[z^{(1)}, z^{(2)}, \dots, z^{(K)}] = \left[\begin{bmatrix} z_C \\ z_M^{(1)} \end{bmatrix}, \begin{bmatrix} z_C \\ z_M^{(2)} \end{bmatrix}, \dots, \begin{bmatrix} z_C \\ z_M^{(K)} \end{bmatrix} \right] \quad (1)$$

As not all paths in Z_M correspond to physically plausible motions, valid paths should be learnt using a recurrent neural network (RNN) R_M . At each time, R_M takes a vector sampled from a Gaussian distribution as input: $\epsilon^k \sim p_E \equiv \mathcal{N}(\epsilon, 0|I_{d_E})$ and outputs a vector in Z_M : $z_M^{(k)} = R_M(k)$. Shortly, the RNN maps a sequence of *random* variables with independent and identical distribution $[\epsilon^{(1)}, \dots, \epsilon^{(K)}]$ to a sequence of *correlated* random variables $[R_M(1), \dots, R_M(K)]$ representing dynamics in a video. In MoCoGAN, R_M is implemented as a one-layer GRU [1].

The framework of MoCoGAN is illustrated in Fig. 2a. It composes of four sub-networks: the recurrent neural network R_M , the image generator G_I , the image discriminator D_I and the video discriminator D_V . The image generator G_I generates a video clip by mapping a vector $\left[\begin{bmatrix} z_C \\ z_M^{(1)} \end{bmatrix}, \begin{bmatrix} z_C \\ z_M^{(2)} \end{bmatrix}, \dots, \begin{bmatrix} z_C \\ z_M^{(K)} \end{bmatrix} \right] \in Z_I$ to a

sequence of images: $\tilde{v} = [\tilde{x}^1, \tilde{x}^2, \dots, \tilde{x}^K]$ where $\tilde{x}^k = G_I(\begin{smallmatrix} z^C \\ z_M^{(k)} \end{smallmatrix})$ and $z_M^{(k)}$ is from $R_M(k)$. D_I and D_V make judges on G_I and R_M . MoCoGAN optimization is as in Eq. (2):

$$\max_{G_I, R_M} \min_{D_I, D_V} F_{mcg1} = \max_{G_I, R_M} \min_{D_I, D_V} \mathcal{F}_V(D_I, D_V, G_I, R_M) \quad (2)$$

where

$$\begin{aligned} \mathcal{F}_V(D_I, D_V, G_I, R_M) = & \mathbb{E}_{\mathbf{v}}[-\log D_I(S_1(\mathbf{v}))] + \mathbb{E}_{\tilde{\mathbf{v}}}[-\log(1 - D_I(S_1(\tilde{\mathbf{v}})))] + \\ & \mathbb{E}_{\mathbf{v}}[-\log D_V(S_T(\mathbf{v}))] + \mathbb{E}_{\tilde{\mathbf{v}}}[-\log(1 - D_V(S_T(\tilde{\mathbf{v}})))] \end{aligned} \quad (3)$$

where $\mathbb{E}_{\mathbf{v}}$ is a shorthand for $\mathbb{E}_{\mathbf{v} \sim p_V}$, and $\mathbb{E}_{\tilde{\mathbf{v}}}$ for $\mathbb{E}_{\tilde{\mathbf{v}} \sim p_{\tilde{V}}}$. S_1, S_T are two random access functions. S_1 takes a video clip and outputs a random frame from the clip while S_T takes a video clip and randomly returns T consecutive frames from the clip. In Eq. (3), the first and the second terms encourage D_I to output 1 for a video frame from a real video clip \mathbf{v} and 0 for a video frame from a generated one $\tilde{\mathbf{v}}$. Similarly, the third and the fourth terms encourage D_V to output 1 for T consecutive video frames from a real video clip \mathbf{v} and 0 for T consecutive frames from a generated one $\tilde{\mathbf{v}}$.

To model categorical dynamic of action/gesture, MoCoGAN utilizes an additional one-hot vector z_A to control the category of action. As a consequence, the objective function changes to \mathcal{F}_{mcg2} Eq. (5):

$$\mathcal{F}_{mcg2} = \mathcal{F}_V(D_I, D_V, G_I, R_M) + \lambda L_I(G_I, Q) \quad (4)$$

$$= \mathcal{F}_{mcg1} + \lambda L_I(G_I, Q) \quad (5)$$

where L_I is a lower bound of the mutual information between the generated video clip and \mathbf{z}_A , Q approximates the distribution of the action category variable conditioning on the video clip which is implemented by adding a softmax layer to the last feature layer of D_V . λ is a hyperparameter, set to 1 in the experiment.

3 vi-MoCoGAN

3.1 Viewpoint Controlling

To generate action/gestures from a certain viewpoint, we introduce an one-hot vector \mathbf{z}_V to control the view of image generator. The fact of introducing \mathbf{z}_V must be carefully considered because it could affect the network's performance. By experiment, we found that the best way is put \mathbf{z}_V into the image generator G_I . Besides, we adapt the objective function to evaluate the view of generated video with the input view control \mathbf{z}_V by a cross-entropy function L_V . In this way, the new objective function \mathcal{F}_{vmcg1} that takes view constraint into account is as Eq. (6):

$$\mathcal{F}_{vmcg1} = \mathcal{F}_{mcg2} + \beta L_V \quad (6)$$

where β is a hyperparameter. We call this model **vi-MoCoGAN v1**.

3.2 Subject Consistency

The current model **vi-MoCoGAN v1** is capable of generating video clip of a certain action at a certain viewpoint. However, in our experiments, we found that it does not ensure well consistency of the subject performing the action. This means **vi-MoCoGAN v1** could generate videos whose the subject is a combination of several body parts of different subjects in the training set. The reason is that in MoCoGAN, there is not an explicit constraint to keep subject’s consistency. In fact, the output of video discriminator D_V in MoCoGAN has only two components: one is a binary value (0 or 1) to generally judge the true video from the generated one, another is a vector to measure the similarity of action category. As a result, it is too weak to judge separately content, action and view of the generated video.

To overcome this limitation, firstly, we do not use a random content vector z_C to input into the generator G_I . Instead, we randomly extract one image I from a given video and pass it through an Image Encoder sub-network to generate the vector $z_C = I_E(I)$. Then, we add a cross-entropy function L_C , that evaluates the similarity of the subject in generated video clip with z_C , in the objective function. The final objective function \mathcal{F}_{vmcg2} of our **vi-MoCoGAN v2** is as Eq. (7):

$$\mathcal{F}_{vmcg2} = \mathcal{F}_{mcg1} + \beta L_V + \gamma L_O = \mathcal{F}_{vmcg1} + \gamma L_O \quad (7)$$

where γ is a hyperparameter. Figure 2 and Table 1 show the difference between the original MoCoGAN and the proposed vi-MoCoGAN.

3.3 Network Architectures

vi-MoCoGAN v1 composes of four sub-networks R_M, G_I^1, D_I^1, D_V^1 similar to the four subnets of MoCoGAN but their inputs and outputs change (see Table 1). Particularly, the output of D_V^1 in **vi-MoCoGAN v1** has $1 + d_A + d_V$ values while the output of D_V in MoCoGAN has only $1 + d_A$ values. The d_V additional values are used to evaluate the view of generated video with the view controlling vector z_V by the entropy function L_V (Eq. (8)).

vi-MoCoGAN v2 composes of five sub-networks: four subnets R_M, G_I^2, D_I^2, D_V^2 and an additional Image Encoder I_E to generate content vector z_C . It notices that the output D_V^2 of **vi-MoCoGAN v2** has $1 + d_A + d_V + d_C$ components, the d_C additional values are used to evaluate the similarity of subject in the generated video with z_C by the entropy function L_C (Eq. (9)). $ExtractV()$ and $ExtractCO()$ are two functions that extract d_V and d_O values corresponding to the distribution of view and content from output of $D_V^{\{1,2\}}$.

$$L_V = CrossEntropy(z_V, ExtractV(d_V, D_V^{\{1,2\}})) \quad (8)$$

$$L_O = CrossEntropy(z_O, ExtractC(d_O, D_V^2)) \quad (9)$$

where z_O and z_V are one-hot vectors indicating the subject and the view to be learnt.

Architectures of sub-networks $R_M, D_I^{\{1,2\}}, D_V^{\{1,2\}}, G_I^{\{1,2\}}$ in vi-MoCoGAN are quite similar to the ones of MoCoGAN, but their configurations are modified to adapt with more inputs and outputs. In most cases, we add an additional layer in each sub-network and we adapt the kernel size, stride and padding. Compared to MoCoGAN, vi-MoCoGAN v2 has more than one sub-network which is Image Encoder I_E . We have designed I_E to generate the content vector z_C and a network Evaluator E to evaluate the quality of the generated videos. Table 2 shows configuration of these sub-networks where d_V is the number of views, d_O is the number of subjects, d_A is the number of action categories, $d_{AVO} = (d_A + d_V + d_O)$, $d_{VO} = (d_V + d_O)$. Hyper-parameters α, β, γ are set to 1. In these tables, N stands for output channels, K, S, P stand for kernel size, stride and padding. Similar to MoCoGAN, we also used ADAM [3] for training, with a learning rate of 0.0002 and momentums of 0.5 and 0.999.

Table 1. Main differences between MoCoGAN and vi-MoCoGAN

Method	Sub-nets	Inputs	#Outputs D_I	#Output D_V	Loss function
MoCoGAN	$R_M, G_I,$ D_I, D_V	z_C, z_A	1	$1 + d_A$	$F + L_I$
vi-MoCoGAN v1	$R_M, G_I,$ D_I^1, D_V^1	z_C, z_A, z_V	$1 + d_V$	$1 + d_A + d_V$	$F + L_I + L_V$
vi-MoCoGAN v2	$R_M, G_I, D_I^2,$ D_V^2, I_E	$z_C = I_E(I), z_A, z_V$	$1 + d_V + d_O$	$1 + d_A + d_V + d_O$	$F + L_I + L_V + L_C$

In original MoCoGAN, all generated video clips has resolution of 64×64 . As these videos contain human action, this resolution is acceptable to recognize the action. However, in our work, we generate videos of human hand gestures where human body do not change so much, only arm and hand posture are changing. However, hand has usually very low resolution, if we generate low resolution video clip, the detail of hand should be lost. In this work, we generate (*height* \times *width*) video clips (128×128 in our experiment).

4 Experiments

4.1 Building the Dataset of Hand Gestures at Different Viewpoints

As mentioned previously, there does not exist a multi-view hand gestures dataset for human machine interaction. Therefore, we have collected a new dataset of twelve dynamic hand gestures, performed by six subjects observed by five cameras. The cameras are uniformly spacing. During gesture implementation, subject stand in front of the third camera at a distance of 2.0 m (Fig. 3). Five cameras (K_1, K_2, K_3, K_4, K_5) are set at the height of 1.5 m at angles ($-90^\circ, -45^\circ, 0^\circ, 45^\circ, 90^\circ$) compared to human orientation. The K_1, K_5 at $-90^\circ, 90^\circ$ are two most difficult views because the human hand is easily occluded by human body. The

Table 2. Configuration of sub-networks in vi-MoCoGAN

G_I	Configuration
Input	$z_M \sim R_M, z_C \sim I_E(I) \in R^{d_C}, z_V \in R^{d_V}$
1	DCONV-(N(512), K(4,4), S(1, 1), P(0, 0)), BN, ReLU
2	DCONV-(N(256), K(4,4), S(2, 2), P(1, 1)), BN, ReLU
3	DCONV-(N(128), K(4,4), S(2, 2), P(1, 1)), BN, ReLU
4	DCONV-(N(128), K(4,4), S(2, 2), P(1, 1)), BN, ReLU
5	DCONV-(N(64), K(4,4), S(2, 2), P(1, 1)), BN, ReLU
6	DCONV-(N(3), K(4,4), S(2, 2), P(1, 1)), Tanh
D_I	Configuration
Input	$height \times width \times 3$
1	CONV-(N(64), K(4,4), S(2, 2), P(1,1)), LeakyReLU
2	CONV-(N(128), K(4,4), S(2,2), P(1,1)), BN, LeakyReLU
3	CONV-(N(128), K(4,4), S(2,2), P(1,1)), BN, LeakyReLU
4	CONV-(N(256), K(4,4), S(2,2), P(1,1)), BN, LeakyReLU
5	CONV-(N(512), K(4,4), S(2,2), P(1,1)), BN, LeakyReLU
6	CONV-(N(1 + d_{VO}), K(4; 4), S(1; 1), P(0; 0)), <i>BN, LeakyReLU</i>
D_V	Configuration
Input	$16 \times height \times width \times 3$
1	CONV3D-(N(64), K(4,4,4), S(1,2, 2), P(0,1,1)), LeakyReLU
2	CONV3D-(N(128), K(4,4,4), S(1,2,2), P(0,1,1)), BN, LeakyReLU
3	CONV3D-(N(128), K(3,4,4), S(1,2,2), P(1,1,1)), BN, LeakyReLU
4	CONV3D-(N(256), K(4,4,4), S(1,2,2), P(0,1,1)), BN, LeakyReLU
5	CONV3D-(N(512), K(4,4,4), S(1,2,2), P(0,1,1)), BN, LeakyReLU
6	CONV3D-(N(1 + d_{AVO}), K(4; 4; 4), S(1; 1; 1), P(0; 0; 0)), <i>BN, LeakyReLU</i>
I_E	Configuration
Input	$height \times width \times 3$
1	CONV-(N(64), K(4,4), S(2,2), P(1,1)), LeakyReLU
2	CONV-(N(128), K(4,4), S(2,2), P(1,1)), BN, LeakyReLU
3	CONV-(N(128), K(4,4), S(2,2), P(1,1)), BN, LeakyReLU
4	CONV-(N(256), K(4,4), S(2,2), P(1,1)), BN, LeakyReLU
5	CONV-(N(512), K(4,4), S(2,2), P(1,1)), BN, LeakyReLU
6	CONV-(N(d_C); K(4; 4), S(1; 1), P(0; 0)), <i>BN, LeakyReLU</i>
E	Configuration
Input	$16 \times height \times width \times 3$
1	CONV3D-(N(64), K(4,4,4), S(1, 2, 2), P(0, 1, 1)), LeakyReLU
2	CONV3D-(N(128), K(4,4,4), S(1, 2, 2), P(0, 1, 1)), BN, LeakyReLU
3	CONV3D-(N(128), K(3,4,4), S(1, 2, 2), P(1, 1, 1)), BN, LeakyReLU
4	CONV3D-(N(256), K(4,4,4), S(1, 2, 2), P(0, 1, 1)), BN, LeakyReLU
5	CONV3D-(N(512), K(4,4,4), S(1, 2, 2), P(0, 1, 1)), BN, LeakyReLU
6	CONV3D-(N(d_{AVO}), K(4; 4; 4), S(1; 1; 1), P(0; 0; 0)), <i>BN, LeakyReLU</i>

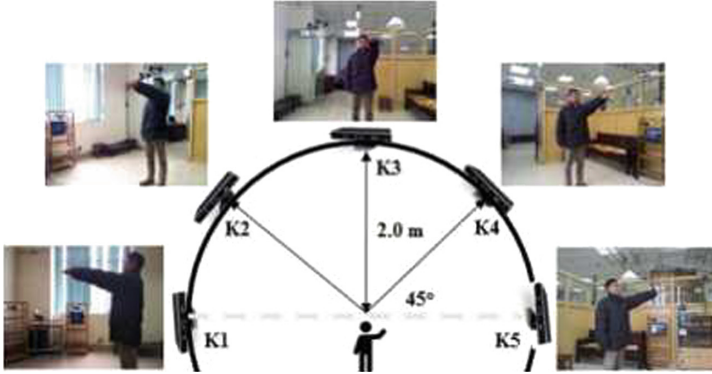


Fig. 3. Setup equipment for collecting multi-view hand gestures

video are captured at 640×480 , 30 fps, we re-sample videos every three frames and resize frames to 128×128 .

In our experiment, each gesture corresponds to a movement of the hand such as up/down, left/right, left circle, right circle, etc. with hand shape following a cycle from closed palm to open palm (open two fingers) then closed palm. The gestures are designed to correspond to some commands controlling home appliances in a smart home. Totally, we have 1080 video samples ($12 \text{ gestures} \times 6 \text{ subjects} \times 5 \text{ views} \times 3 \text{ realization times}$) for training the vi-MoCoGAN v1 or v2. It notices that there are two pairs of gestures whose trajectories of the hand are similar, the main difference is only in the hand posture (open palm or open two fingers of the hand) so if the generated video has low resolution, it could be hard to recognize this difference. In our experiments, d_C is set to 50 and d_M is set to 10 as in MoCoGAN, d_V the number of views is set to 5, d_A is 12, d_O the number of subjects in the experiments.

4.2 Evaluation Metrics

As mentioned in [8], evaluating generative models is known to be a challenging task. There are not common metrics for evaluating generated videos. In MoCoGAN, the authors utilized Average Content Distance (ACD) and Motion Control Score (MCS). The higher ACD shows the better content consistency of the generated video while the MCS shows the capacity in motion generation control. To compute ACD, depending on the dataset that the authors measure the L_2 pairwise distance between two consecutive frames with the features computed using OpenFace for face dataset. To compute MCS, the authors trained a classifier to recognize the generated actions. The better accuracy of the classifier, the higher generated video quality.

In this work, we propose three evaluation metrics for evaluating video generation results under different viewpoints. We utilize a 3D CNN E that has been learnt from labelled dataset of hand gestures at different viewpoints. Through

the training, we hope E will recognize the most important features of real videos then use it as an arbitrator to judge on the quality of generated videos in term of action, subject and view. As we can see in Table 2, E has the similar architecture of the video discriminator D_V . The output of E is the predicted values of action, subject, view and each represented as an one-hot vector.

We assume q_A, q_O, q_V are one-hot vectors to input to vi-MoCoGAN to generate video clip for gesture A , subject O and view V respectively. The generated video will be input to the network E and outputs three vectors p_A, p_O, p_V predicting the gesture, the subject and the view of the generated video. We then compare the similarity of the truth information with the predicted one by CrossEntropy function. So we have three metrics: Object Control Score (OCS); View Control Score (CVS) and Motion Control Score (MCS). The higher values of OCS, VCS and MCS, the better quality of generated video.

$$OCS = CrossEntropy(p_O, q_O) \quad (10)$$

$$VCS = CrossEntropy(p_V, q_V) \quad (11)$$

$$MCS = CrossEntropy(p_A, q_A) \quad (12)$$

4.3 Experimental Results

This section shows the evaluation results obtained by **vi-MoCoGAN v1** (MoCoGAN with view control - Sect. 3.1) and **vi-MoCoGAN v2** (MoCoGAN with view control and subject consistency - Sects. 3.1 and 3.2). We start first by giving some qualitative results showing the limitation of **vi-MoCoGAN v1** and improvement made by **vi-MoCoGAN v2**.

Qualitative Evaluation. We evaluate qualitatively the generated videos by **vi-MoCoGAN v1** and **vi-MoCoGAN v2** in term of subject, view and action.

Subject Consistency Evaluation. Figure 4 shows the key-frames of two videos generated by **vi-MoCoGAN v1** (first row) and **vi-MoCoGAN v2** (second row). We observe that at the first row, **vi-MoCoGAN v1** generates video with poor quality in term of subject consistency. Most of frames in this video contains the same subject but in several frames (e.g second frame) contains another subject. The quality of generated frames is poor. There is many shadow appeared in the frames so we can not see clearly the subject and the performed gesture. In contrast with **vi-MoCoGAN v1**, **vi-MoCoGAN v2** gives significantly better video quality. The subject remains the same during gesture implementation. The quality of frames is better and we can observe clearly the gesture performed by the subject through consecutive frames.

View Consistency Evaluation. Figure 5 shows key-frames of two videos generated by **vi-MoCoGAN v1** (first row) and **vi-MoCoGAN v2** (second row) for the fourth view (K_4). In this example, we observe that the video generated by

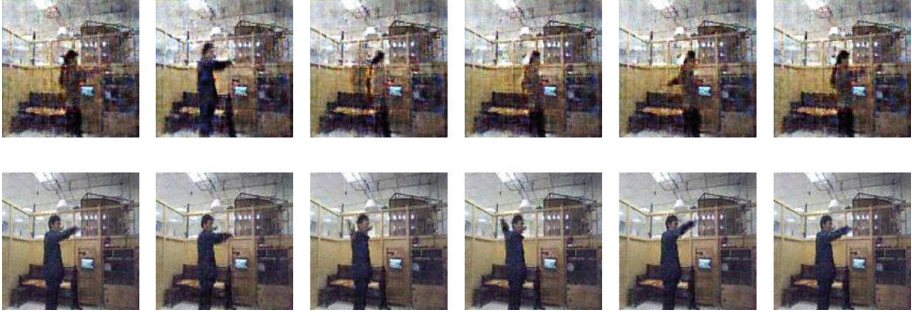


Fig. 4. Evaluation of videos generated by **vi-MoCoGAN v1** (first and third rows) and **vi-MoCoGAN v2** (second and fourth rows) in term of subject consistency.

vi-MoCoGAN v1 (first row) has very low quality. Mainly the gesture is generated at view K_4 but we observe also appearance of the subject at view K_1 . Conversely, this drawback has been resolved with **vi-MoCoGAN v2** (second row). This time, the view of gesture is correctly generated (the fourth view K_4). This shows that it seems that with the control of view, the network could generate the video at correct view. But this depends strongly on the subject consistency. If the subject consistency is well controlled, the view problem could be resolved.



Fig. 5. Evaluation of videos generated by **vi-MoCoGAN v1** (first row) and **vi-MoCoGAN v2** (second row) in term of view consistency.

Action Evaluation. Figure 6 shows key-frames of four videos (two videos generated by **vi-MoCoGAN v1** (first and third rows) and two videos generated by **vi-MoCoGAN v2** (second and fourth rows)) for two subjects at two views K_5 (-90°) and K_3 (0°). Once again, we observe that the quality of the videos generated by **vi-MoCoGAN v1** is very poor and it is very hard to recognize the gestures even by human eyes. **vi-MoCoGAN v2** performs better and generates videos in which we can recognize the pre-defined gestures.

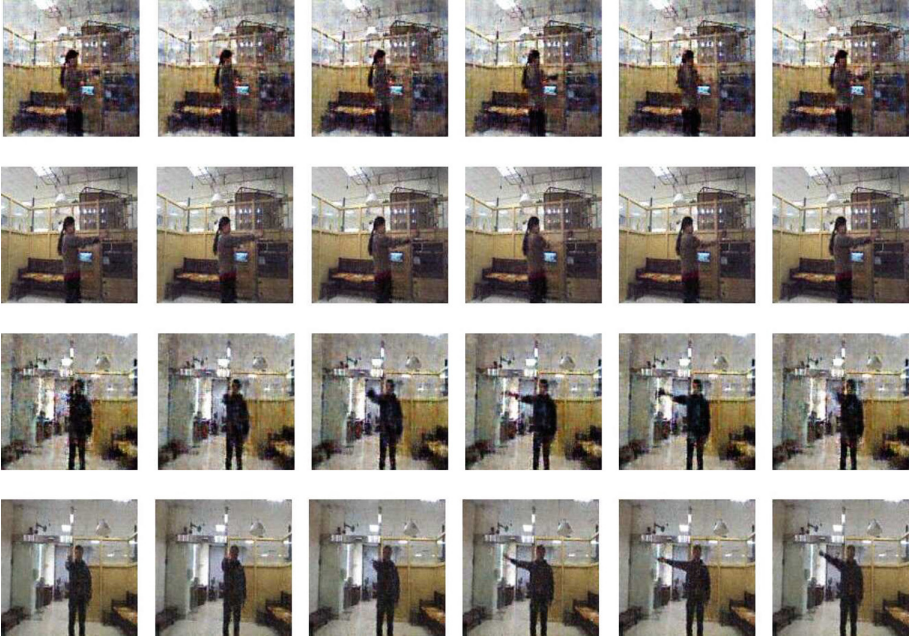


Fig. 6. Evaluation of videos generated by **vi-MoCoGAN v1** (first row) and **vi-MoCoGAN v2** (second row) in term of gestures

Quantitative Evaluation. We have trained the evaluation network E with a subset of multi-view hand gestures performed by two subjects, five views, four gestures each realized 3 times. Totally we have 120 videos for training. We generate 40 videos ($5 \text{ views} \times 2 \text{ subject} \times 4 \text{ gestures}$) by **vi-MoCoGAN v1** and 40 videos ($5 \text{ views} \times 2 \text{ subject} \times 4 \text{ gestures}$) by **vi-MoCoGAN v2**.

Table 3. Quantitative evaluation of vi-MoCoGAN

Method/Score	VCS	OCS	ACS
vi-MoCoGAN v1 (MoCoGAN + View control)	0.8	0.65	0.75
vi-MoCoGAN v2 (MoCoGAN + View control + Subject Consistency)	1	0.9	0.99

Table 3 shows that the fact of integrating view and subject constraints in the MoCoGAN allows **vi-MoCoGAN v2** generates better video clip: the subject, view and action are well generated. In term of view, VCS obtained by **vi-MoCoGAN v1** is only 0.8 while this value is 1 by **vi-MoCoGAN v2**. As analysed in the

qualitative evaluation section, the quality of videos generated by vi-MoCoGAN v1 is very poor to distinguish two views. Subject consistency score is very low with vi-MoCoGAN v1 (OCS = 0.65) while vi-MoCoGan v2 has OCS up to 0.9. In term of gestures, vi-MoCoGan v2 has higher action consistency score (0.99) than vi-MoCoGan v1 (0.75).

Figure 7 illustrates the real videos (first row) and the generated videos by of vi-MoCoGAN v2 (second row) of one subject performing fourth gestures at five different views. This figure is best viewed with Acrobat/Foxit Reader on a desktop. The readers are invited to click to images to play the video clip. We can see that compared to the real video, the generated videos have good view-point, subject consistency and the generated gestures are correctly recognized and comparable to the real gestures. Sometimes, the phase of generated gestures changes comparing to the phase of the real gestures but this enrich the gestures set and could help for training recognition model.



Fig. 7. Comparison of real gestures (first row) performed by a subject and the corresponding generated gestures (second row) at five different views (from left to right K_1 (-90°), K_2 (-45°), K_3 (0°), K_4 (45°), K_5 (90°)) (see Supplementary material). The figure is best viewed with Acrobat/Foxit Reader on a desktop.

5 Conclusion

We have presented a variant of MoCoGAN for generating videos under different viewpoints. To the best of our knowledge, this is the first work for video generation at certain viewpoints. The experiments have been conducted with the case of human hand gestures, showing good video quality in term of view, subject and gestures. This is a good step for augmenting data which enriches the current set of data without requiring annotation which is very time consuming. However, this is an ongoing work so many tasks will be conducted in the future. Firstly, we will investigate deeply the role of view and subject controlling for

video generation. Currently, without subject controlling, the view of generated video is not good. The first experiment could be changing the value of hyper-parameters. Secondly, we will evaluate quantitatively the proposed methods for every subject and gesture and test to generate video of other multi-view action datasets. Thirdly, now the videos are generated under the same view and subject in the training set. It would be able to generate videos with new subjects at novel viewpoints. Finally, the generated videos will be used as augmented data for training gesture recognizer.

References

1. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling (2014). arXiv preprint [arXiv:1412.3555](https://arxiv.org/abs/1412.3555)
2. Goodfellow, I., et al.: Generative adversarial nets. In: *Advances in Neural Information Processing Systems*, pp. 2672–2680 (2014)
3. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization (2014). arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980)
4. Nguyen, D.H., Le, T.H., Tran, T.H., Vu, H., Le, T.L., Doan, H.G.: Hand segmentation under different viewpoints by combination of mask R-CNN with tracking. In: *2018 5th Asian Conference on Defense Technology (ACDT)*, pp. 14–20. IEEE (2018)
5. Rautaray, S.S., Agrawal, A.: Vision based hand gesture recognition for human computer interaction: a survey. *Artif. Intell. Rev.* **43**(1), 1–54 (2015)
6. Ruffieux, S., Lalanne, D., Mugellini, E., Abou Khaled, O.: A survey of datasets for human gesture recognition. In: Kurosu, M. (ed.) *HCI 2014*. LNCS, vol. 8511, pp. 337–348. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07230-2_33
7. Saito, M., Matsumoto, E., Saito, S.: Temporal generative adversarial nets with singular value clipping. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2830–2839 (2017)
8. Theis, L., Oord, A.V.D., Bethge, M.: A note on the evaluation of generative models (2015). arXiv preprint [arXiv:1511.01844](https://arxiv.org/abs/1511.01844)
9. Tian, Y., Peng, X., Zhao, L., Zhang, S., Metaxas, D.N.: CR-GAN: learning complete representations for multi-view generation (2018). arXiv preprint [arXiv:1806.11191](https://arxiv.org/abs/1806.11191)
10. Truong, D.M., Doan, H.G., Tran, T.H., Vu, H., Le, T.L.: Robustness analysis of 3D convolutional neural network for human hand gesture recognition. *Int. J. Mach. Learn. Comput.* **8**(2), 135–142 (2019)
11. Tulyakov, S., Liu, M.Y., Yang, X., Kautz, J.: MoCoGAN: decomposing motion and content for video generation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1526–1535 (2018)
12. Villegas, R., Yang, J., Hong, S., Lin, X., Lee, H.: Decomposing motion and content for natural video sequence prediction (2017). arXiv preprint [arXiv:1706.08033](https://arxiv.org/abs/1706.08033)
13. Vondrick, C., Pirsiavash, H., Torralba, A.: Generating videos with scene dynamics. In: *Advances in Neural Information Processing Systems*, pp. 613–621 (2016)

**Image and Pattern Analysis for
Multidisciplinary Computational
Anatomy**



Weakly Supervised Domain Adaptation with Point Supervision in Histopathological Image Segmentation

Shun Obikane and Yoshimitsu Aoki(✉)

Keio University, 3-14-1, Hiyoshi, Kohoku, Yokohama, Kanagawa, Japan
sobikane@aoki-medialab.jp, aoki@elec.keio.ac.jp

Abstract. When a model learned in a domain is applied to a different domain, even if in the same task, there is no guarantee of accuracy. This is a very important issue when deep learning and machine learning are applied in the field. In medical applications, there is a wide variety of domain bias, making it very difficult to create a model appropriate for all domains. Furthermore, semantic segmentation needs fine annotation and its high labor cost makes its application difficult. Histopathological image segmentation enables drug discovery and medical image analysis, but it is expensive due to its annotation cost and the need for the skills of histopathological experts. In this paper, we focus on a weakly supervised method using point annotation unique to histopathological image segmentation, and tackled on weakly supervised domain adaptation to suppress domain gaps. Providing point level annotation instead of fine annotation decreases the high cost of labor normally required.

Keywords: Histopathology image segmentation · Semantic segmentation · Weakly supervised domain adaptation · Medical image analysis

1 Introduction

1.1 Domain Adaptation

Convolutional Neural Networks (CNNs) achieve great success in many tasks such as image classification, object detection, and action recognition. However, CNNs cannot guarantee performance in unseen data because of the variety of environments (domain gaps). Thus, there is a need to annotate the data for new domains and remake the models. There are, however, obstacles that arise. For example, object detection and semantic segmentation require fine annotation, which has a high labor cost. Annotation cost is an important issue when applying machine learning and deep learning to a social problem. The goal is to reduce annotation cost and make proper models in a wide variety of domains. Domain adaptation tackles such problems and aims to reduce domain gaps in training data (source data) and testing data (target data). By using source data that is

fully annotated and target data that is not annotated or not fully annotated (the weakly supervised method), we are able to make an appropriate model in a new domain for less cost. In domain adaptation (DA), supervision is defined by the target data's annotation level; essentially, where the source data is fully annotated and the target data is not, the process is referred to as an unsupervised domain adaptation (UDA). If some target data is annotated, it is considered a semi-supervised domain adaptation (SDA). If the target data is not fully annotated but some weak annotation exists, this is referred to as a weakly-supervised domain adaptation (WDA). Several UDA methods have shown great progress [7, 9, 13, 27, 30]; many such methods have been proposed for semantic segmentation [17, 18, 32, 34, 36]. The UDA method is being suggested as more complex method and the number of hyperparameters required is increasing. Hyperparameter tuning in medical application, which has several domains, is difficult due to the shortage of experts. In this paper we introduce weak annotation into a simple UDA method, creating a WDA. The result is a simple domain adaptation method that guarantees performance in the target data at a lower cost when compared to previous methods.

1.2 Medical Image Analysis

Recently, there have been many studies on medical image segmentation, such as those on histopathological image segmentation [8, 10, 37], MRI tumor image segmentation [21], and retinal vessel image segmentation [12]. These studies achieved significant progress, but wide domain gaps still exist in biomedical image analysis (e.g., camera, organs, staining method). In medical applications, it is particularly necessary to guarantee high performance, so a proper model for each area must be made due to many domain gaps. Semantic segmentation requires fine annotation that has a high labor cost. Experts are needed, making the total cost of annotation higher and creating a serious problem in medical applications.

1.3 Weakly Supervised Semantic Segmentation

Recently, weakly-supervised segmentation methods have been developed [1, 3, 15, 22, 28, 29]. Weak annotation involves items at the image level, point level and at the level of the bounding box that are not fully annotation but provide helpful annotations. The UDA method has progressed, but the number of hyperparameters has increased, complicating the process accordingly. We use weak annotation as the target label and aim to make an easy-to-handle model for medical application. In histopathological image segmentation, point annotation and bounding boxes are primarily used. We used point level annotation from the viewpoint of the fineness of cell size and ease of handling point information in histopathological images. This paper contributes to the literature by applying a WDA to histopathological image segmentation, showing that by using a point-level annotation, which is a low cost construct compared to full annotation, it is possible to improve the accuracy in the target domain by combined it with a

simple UDA method, such that the number of hyperparameters is low and it is an easy-to-handle model.

2 Related Work

2.1 Histopathological Image Segmentation

Many histopathology image segmentation methods have been developed [4, 5, 19, 25, 37]. Semantic segmentation addresses a wide variety of issues unique to histopathology image segmentation. [25] treats regression in cell images, and [37] focuses on a cell segmentation problem that requires finer classification. [14] is weakly supervised method with point annotation. It processed pseudo-labels by combining k-means clustering and Conditional Random Fields (CRF).

2.2 Medical Image Domain Apdaptation

Domain adaptation in medical image analysis has progressed [6, 16, 20, 33]. In many cases, there are multiple methods available to obtain common domain representation to solve domain gaps. [20] deals with pneumonia classification problems. This study uses Generative Adversarial Network (GAN), which generates images such that it is difficult to discriminate between the source and the target, so the classification model is used for a common domain. Domain adaptation is also progressing in the area of histopathological image segmentation. [6] prepared models for each source and target and used maximum mean discrepancy (MMD) or correlation alignment (CORAL), which measure the difference between feature distributions in each model as a loss function to resolve domain discrepancy. [33] transferred the source image to the target style by using Cycle-GAN to solve domain gaps in image style using train data. [16] used an adversarial learning method where the common segmenter and discriminator were provided. The discriminator decides which domain is input from the common segmenter output, and common domain representation is obtained.

3 Weakly Supervised Domain Adaptation

3.1 Unsupervised Domain Adaptation

As an introduction to our method, UDA is explained. In UDA, it has been experimentally shown that adversarial learning is effective, and many methods have adopted it [11, 17, 30, 34, 36]. These methods commonly set the discriminator, which distinguishes whether input data is a source or a target and solves adversarial loss L_{adv} , so generater get proper model for both source and target domain. Figure 1 shows an overview of our method networks. The segmenter \mathbf{G} output is the segmentation result. Based on hidden layer outputs, discriminator \mathbf{D} distinguishes whether input data is a source image $\mathbf{I}_s \in \mathbb{R}^{(H \times W \times 3)}$ (fully

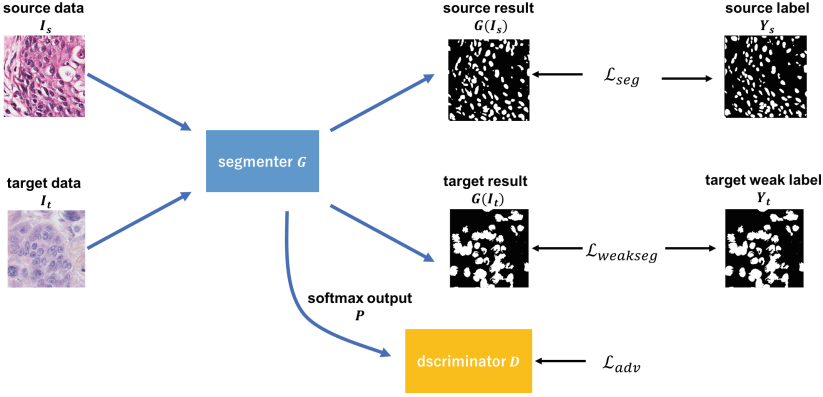


Fig. 1. An overview of our method. Segmenter \mathbf{G} outputs the segmentation result. Discriminator \mathbf{D} distinguishes whether the segmenter’s softmax output is from source data $\mathbf{I}_s \in \mathbb{R}^{(H \times W \times 3)}$ or target data $\mathbf{I}_t \in \mathbb{R}^{(H \times W \times 3)}$. Then the adversarial loss is optimized \mathcal{L}_{adv} and the segmenter is given a good model for both domains, with which the discriminator can determine whether an input is from the source or target. In UDA, segmentation loss is only \mathcal{L}_{seg} on source data, but in WDA, the weakly segmentation loss $\mathcal{L}_{weakseg}$ on target data is added to this.

annotated by $\mathbf{Y}_s \in \mathbb{R}^{(H \times W)}$ or a target image $\mathbf{I}_t \in \mathbb{R}^{(H \times W \times 3)}$ (not annotated). Domain adaptation for semantic segmentation [32] shows that low-dimensional softmax output $\mathbf{P} = \mathbf{G}(\mathbf{I}) \in \mathbb{R}^{(H \times W \times C)}$, where C is the number of categories, is better for discriminator input than high-dimensional hidden layer outputs, so this was adopted for this study. While segmenter outputs are difficult for the discriminator to distinguish the domain of, in this case, the discriminator learns from the segmenter outputs which domain they come from. Thus, after adversarial learning, such an adapted segmenter matches feature distributions between source and target. So, UDA scheme can be written as follows.

Segmenter Training. We define the segmentation loss in (1) as the cross-entropy loss for source data $\{\mathbf{I}_s, \mathbf{Y}_s\}$:

$$L_{seg}(\mathbf{I}_s) = - \sum_{h,w} \sum_{c \in C} \mathbf{Y}_s^{(h,w,c)} \log \mathbf{P}_s^{(h,w,c)} \quad (1)$$

Discriminator Training. As discriminator input, we use segmenter softmax output $\mathbf{P} = \mathbf{G}(\mathbf{I}) \in \mathbb{R}^{(H \times W \times C)}$. And to train discriminator, we use discriminator loss in (2) cross-entropy loss L_D for two classes (source and target). So, it can be written as:

$$L_D(\mathbf{P}) = - \sum_{h,w} (1-z) \log(\mathbf{D}(\mathbf{P}^{(h,w,0)})) + z \log(\mathbf{D}(\mathbf{P}^{(h,w,1)})) \quad (2)$$

where $z = 0$ if input data is draw from target domain, and $z = 1$ if ininput data from source domain.

Adversarial Learning. For target data, to make target prediction distribution $\mathbf{P}_t = \mathbf{G}(\mathbf{I}_t) \in \mathbb{R}^{(H \times W \times C)}$ close to source prediction distribution \mathbf{P}_s , we use adversarial loss L_{adv} in (3) written as:

$$L_{adv}(\mathbf{I}_t) = - \sum_{h,w} \log(\mathbf{D}(\mathbf{P}_t^{(h,w,1)})) \quad (3)$$

So, we formulate objective function for domain adaptation:

$$L(\mathbf{I}_s, \mathbf{I}_t) = L_{seg}(\mathbf{I}_s) + \gamma L_{adv}(\mathbf{I}_t). \quad (4)$$

And optimizing min-max criterion:

$$\max_{\mathbf{D}} \min_{\mathbf{G}} L(\mathbf{I}_s, \mathbf{I}_t), \quad (5)$$

we aim to maximize the probability of predictions in target data while minimizing segmentation loss in source data. By optimizing min-max criterion 5, the segmenter gets a common representation that solves the domain gaps.

3.2 Weakly Supervised Domain Adaptation

There are many weakly-supervised annotations. Image level annotation is given only object identification, point annotation is given object position, bounding box is given object rectangles and so on. For this paper, point annotation was determined to be best in histopathological image segmentation because of its fineness in a large number of cells. In addition, as shown in Fig. 2, we experimented with three types of weak labels: point level annotation, gaussian level annotation, and superpixel level annotation.

Point Level Annotation. Point level annotation give information by points to each cells. In this paper, this weakly label expresses as point level weakly annotations.

Point Annotation with Gaussian Function (Gaussian Level). In addition to point level annotations, we give gaussian level annotation which gaussian functions are center at each point annotations. In this paper, this weakly label expresses as gaussian level weakly annotations.

Point Annotation with Superpixel (Superpixel Level). First, images is divided into superpixel (we used SLIC algorithm [2]), and gives annotations to superpixel which is given point level annotations. In this paper, this weakly lable expresses as superpixel level weakly annotations.

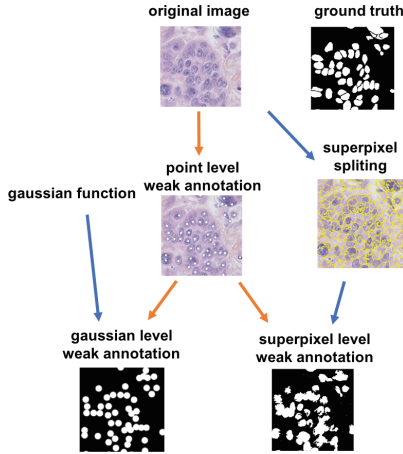


Fig. 2. An overview of the weakly labeling method on target data. A point level annotation is given to each nuclei in the image. This is point level weakly annotation. In addition, we used gaussian level annotation; gaussian functions are centered at each point annotation. This is called gaussian level weakly annotations. Superpixel level annotation is when images are divided into superpixels and annotations are given to the superpixel at the point level annotations. This is called superpixel level weakly annotation.

Segmentation Loss with Weakly Label. In weakly supervised segmentation, [31] says partial cross entropy loss which uses only labeled points $\mathbf{p} \in \Omega_L$ with ground truth is effective. So, we adopted it in our method.

$$L_{weakseg}(\mathbf{I}_t) = - \sum_{\mathbf{p} \in \Omega_L} y_i \log p \quad (6)$$

We add weakly-segmentation loss to unsupervised domain adaptation loss function (5). Thus we optimize weakly domain adaptation loss function (7).

$$L(\mathbf{I}_s, \mathbf{I}_t) = L_{seg}(\mathbf{I}_s) + \gamma_1 L_{weakseg}(\mathbf{I}_t) + \gamma_2 L_{adv}(\mathbf{I}_t)$$

$$\max_{\mathbf{D}} \min_{\mathbf{G}} L(\mathbf{I}_s, \mathbf{I}_t) \quad (7)$$

4 Experiments

4.1 Dataset

Source Data. The source data is the Monuseg dataset [24]. The dataset consists of annotated hematoxylin and eosin (H&E) stained histology images captured at 40 x magnification and made available by the Indian Institute of Technology,

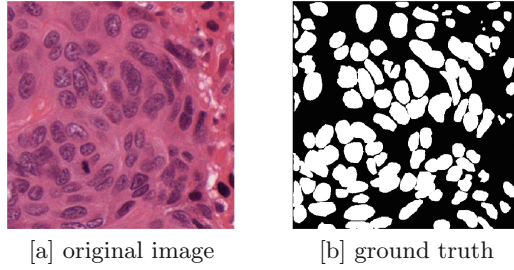


Fig. 3. MoNuseg dataset

Guwahati. This dataset was selected from several patients at several hospital and was extracted in a 1000×1000 patch. There are seven cancer types. An example is shown in Fig. 3. This dataset consists of 30 images and 21623 nuclei are annotated (Fig. 4).

Target Data. The target data is the TNBC dataset [23]. This dataset is annotated H&E stained histology images captured at $40 \times$ magnification and made available by the Curie Institute. All slides are taken from a cohort of Triple Negative Breast Cancer (TNBC) patients and were scanned with a Philips Ultra Fast Scanner 1.6RA. For eleven patients, we extracted 512×512 patches from different areas of tissue. This dataset consists of 50 images and 4022 nuclei are annotated. Additionally, this dataset has been annotated by three experts, guaranteeing its annotation level. In this paper, in order to evaluate them in tandem with the target data, the 50 images were divided into two groups so they could be evaluated in a 2-fold cross validation.

4.2 Experiment Conditions

Segmenter Network and Pre-training. As segmenter model, we used drc-26 [35] which has dilated convolution and pre-trained on ImageNet. To pre-train segmenter for L_{seg} in (1), we use source data $\{I_s, Y_s\}$ and used Adam optimizer with learning rate 1.10^{-2} .

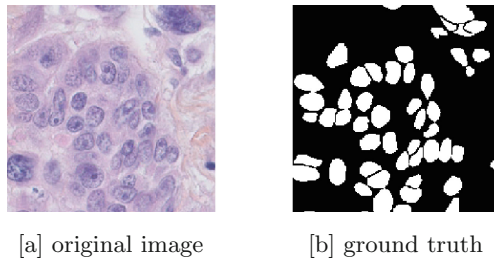


Fig. 4. TNBC dataset

Discriminator Network. As discriminator network, we use architecture similar to [32]. It consists of 5 convolution layers (kernel size is 4×4 and stride is 2), and channel number is {64, 128, 256, 512, 1}. Except for the last layer, a leaky ReLU parameterized by 0.2 and batch normalization follows in each convolution layer.

Network Training in Domain Adaptation. In all experiments we set batch size to 8 and random crop (512×512 in only source data), random 4 rotation 90 degrees for data augmentation. To train segmenter, we used Adam optimizer with learning rate 1.10^{-4} . And to train discriminator, We used the momentum SGD optimizer with (momentum is 0.9 and weight decay is 5.10^{-4}). The learning rate is decreased with the polynomial decay with power of 0.9. For γ_1 and γ_2 , the optimum parameters were selected in the range of 0.01 to 0.5 respectively. We implement our network using the PyTorch toolbox on a single NVIDIA GeForce GTX 1080 Ti GPU. All source data 50 images were used as train data. Target is divided into two groups for 2-fold cross validation, so finally their score is averaged.

4.3 Results

The results are shown in Table 1. These were evaluated by foreground intersection-over-union (fIoU) and F-measure. The experimental conditions for the comparative experiment follows.

Source Model. The source model is learned by using only 50 images from the source data.

Target Model. The target model is learned by using only 25 images from the target data. Domain adaptation aims at this value. Table 1 represents the differences of fIoU in the target model as domain gaps.

DA only (unsupervised DA). This is the unsupervised domain adaptation result. The source data with full annotation and the target data with no annotation were used as training data.

Point Level (weakly Supervised DA). This is the weakly supervised domain adaptation result. Source data with full annotation and target data with point level weakly annotation were used as training data.

Gauss Level (weakly Supervised DA). This is the weakly supervised domain adaptation result. Source data with full annotation and target data with gaussian level weakly annotation were used as training data.

Superpixel Level (weakly Supervised DA). This is our proposed weakly supervised domain adaptation result. Source data with full annotation and target data with superpixel level weakly annotation were used as training data.

Overall Results. Table 1 is list of evaluation values of floU, F measure, pixel accuracy, and floU gap which shows difference from target. In domain adaptation, target model result is the upper-bound result. So, in this experiment, the upper bound is 0.682. For WDA with superpixel level annotation, although the floU gap is 0.154, it has been reduced significantly domain gaps compared to other methods.

Figure 5 shows the output results of the method used in these comparative experiments. Looking at the output results, the source model does not give the target information well, so there are many misidentified areas in which the annotation is not given. Although the results of the unsupervised domain adaptation have been improved, it was not possible to sufficiently reduce mis-recognition. Our method, given the superpixel weakly labeling, can cause a reduction to a level that can be mis-recognized. On the other hand, when compared to weakly supervised methods, the result of point level annotation is the same as in an unsupervised method. Gaussian level annotations are an improvement, but the superpixel level is the best. Thus, it is important to give weakly annotations that capture a certain shape.

Figure 6 is the output result of grad-CAM [26], which visualizes where the discriminator focuses. The source data’s result remains unchanged because the source data is fully annotated. It appears that the discriminator focuses on the object area of the segmenter output and so tends to judge the target result using the worse result and output good result for target data.

Table 1. List of evaluation values of floU and F measure. The difference from the target model is shown as the floU gap.

Annotation level	fIoU	F-measure	Pixel accuracy	fIoU gap
Base model				
Source model	0.441	0.584	0.893	-0.241
Target model	0.682	0.822	0.956	-
Unsupervised				
DA only	0.472	0.611	0.909	-0.210
Weakly supervised				
Point level	0.495	0.648	0.933	-0.187
Gaussian level	0.506	0.646	0.934	-0.176
Superpixel level	0.528	0.684	0.937	-0.154

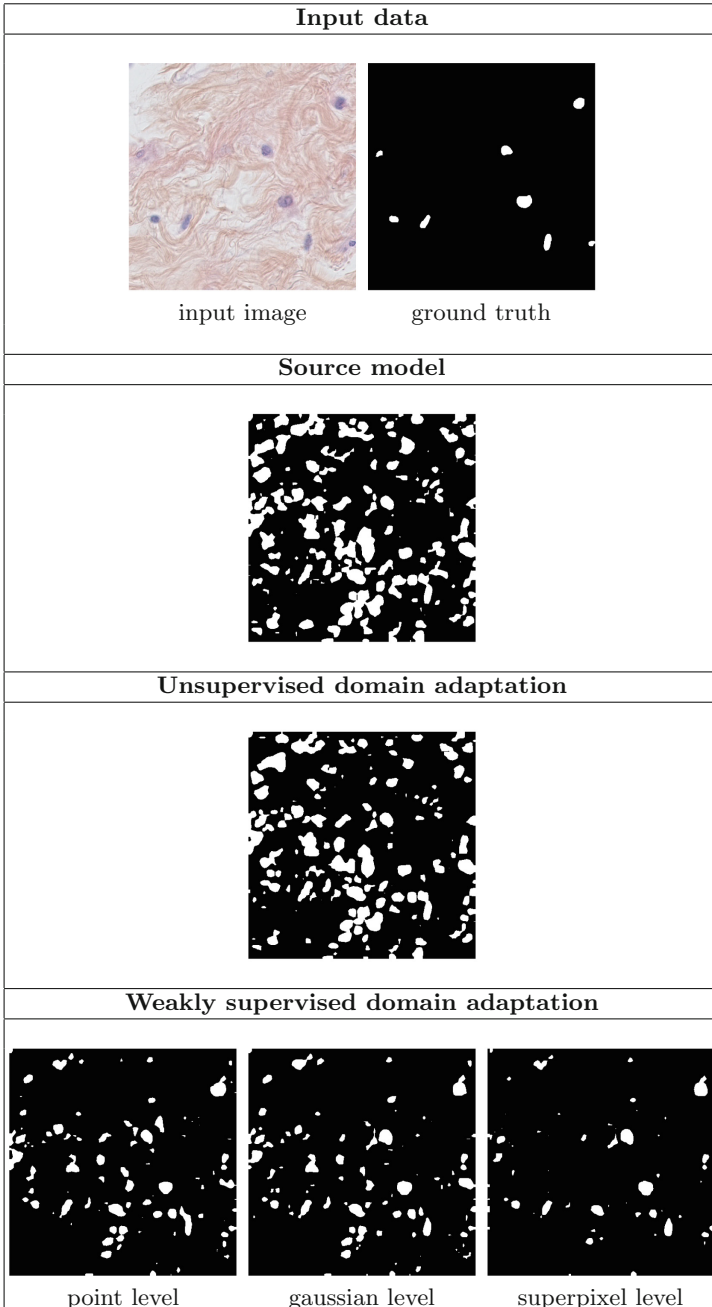


Fig. 5. Output results. The top shows the input data and the ground truth. Next is the result of the source model, which is trained-only source data. Third from the top is the result of unsupervised domain adaptation, and the bottom is the result of weakly supervised domain adaptation with point level, gaussian level, and superpixel level.

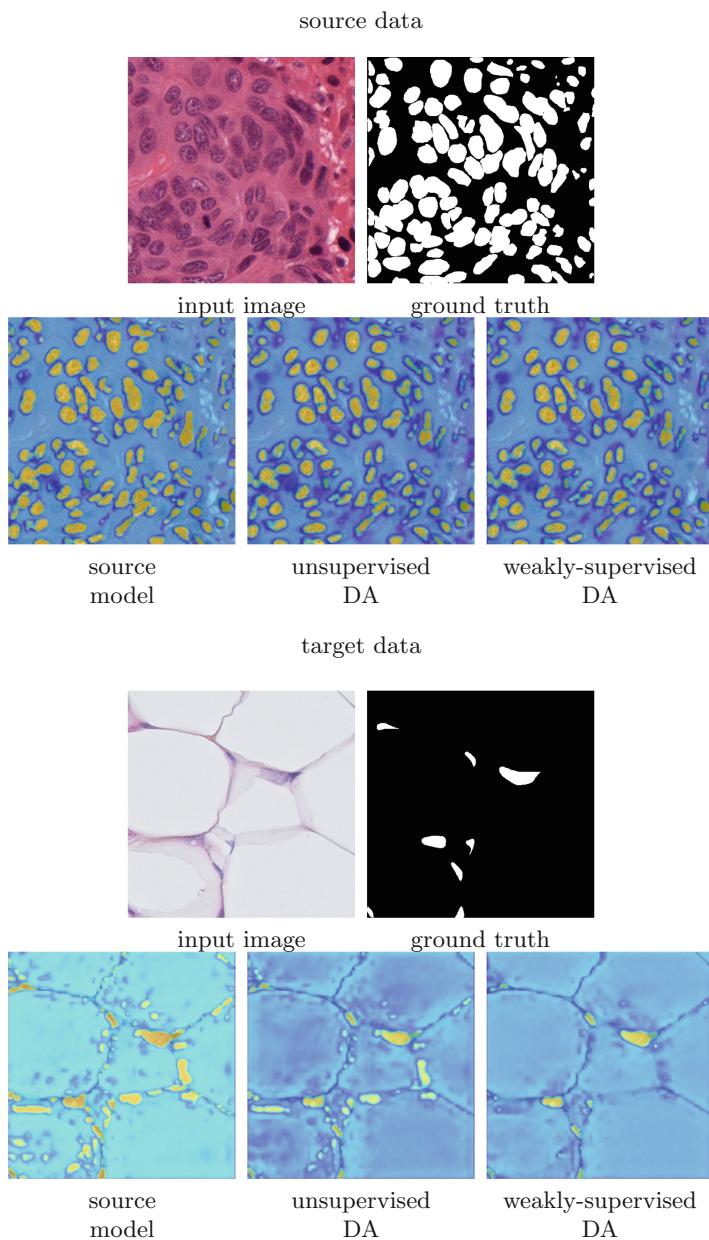


Fig. 6. Output result of grad-CAM [26], which visualizes the focus of the discriminator. The yellow region indicates a larger value and represents where the discriminator looks to distinguish input data domain.

5 Conclusion

In this paper, we showed that weakly supervised domain adaptation is useful in histopathological image segmentation. Our method combines a simple unsupervised domain adaptation method with weak labeling. In the weak label method, the image is divided into superpixels and annotations are given to the superpixels at the point level annotations. The experiments show that this method resolves domain gaps construct to unsupervised domain adaptation and shows the effectiveness of weakly annotation. In the future, we hope to combine weakly-supervised semantic segmentation method.

References

1. Bearman, A., Russakovsky, O., Ferrari, V., Fei-Fei, L.: What's the point: semantic segmentation with point supervision. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9911, pp. 549–565. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46478-7_34
2. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S.: SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(11), 2274–2282 (2012)
3. Khoreva, A., Benenson, R., Hosang, J., Hein, M., Schiele, B.: Simple does it: weakly supervised instance and semantic segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 876–885 (2017)
4. Akram, S.U., Kannala, J., Eklund, L., Heikkilä, J.: Cell proposal network for microscopy image analysis. In: *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 3199–3203, September 2016. <https://doi.org/10.1109/ICIP.2016.7532950>
5. Chidester, B., Ton, T.V., Tran, M.T., Ma, J., Do, M.N.: Enhanced rotation-equivariant U-Net for nuclear segmentation. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019
6. Bermúdez-Chacón, R., Márquez-Neila, P., Salzmann, M., Fua, P.: A domain-adaptive two-stream u-net for electron microscopy image segmentation. In: *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pp. 400–404, April 2018. <https://doi.org/10.1109/ISBI.2018.8363602>
7. Lee, C.Y., Batra, T., Baig, M.H., Ulbricht, D.: Sliced wasserstein discrepancy for unsupervised domain adaptation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10285–10295 (2019)
8. Arbel, E., Remer, I., Ben-Dor, A.: Deep learning based partial annotation framework for instance segmentation in histopathology images (2019)
9. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7167–7176 (2017)
10. Simon, G. et al.: Hover-Net: simultaneous segmentation and classification of nuclei in multi-tissue histology images (2019). arXiv preprint [arXiv:1812.06499v4](https://arxiv.org/abs/1812.06499v4)
11. Ganin, Y., et al.: Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **17**(59), 1–35 (2016). <http://jmlr.org/papers/v17/15-239.html>
12. Fu, H., Xu, Y., Wong, D.W.K., Liu, J.: Retinal vessel segmentation via deep learning network and fully-connected conditional random fields. In: *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, pp. 698–701. IEEE (2016)

13. Laradji, I.H., Babanezhad, R.: M-ADDA: unsupervised domain adaptation with deep metric learning (2018). ArXiv abs/1807.02552
14. Qu, H., et al.: Weakly supervised deep nuclei segmentation using points annotation in histopathology images. In: International Conference on Medical Imaging with Deep Learning - Full Paper Track, London, United Kingdom, 08–10 July 2019. <https://openreview.net/forum?id=H1xkWv8gx4>
15. Laradji, I.H., Rostamzadeh, N., Pinheiro, P.O., Vazquez, D., Schmidt, M.: Where are the blobs: counting by localization with point supervision. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 547–562 (2018)
16. Javanmardi, M., Tasdizen, T.: Domain adaptation for biomedical image segmentation using adversarial training. In: 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), pp. 554–558, April 2018. <https://doi.org/10.1109/ISBI.2018.8363637>
17. Hoffman, J., et al.: CyCADA: cycle-consistent adversarial domain adaptation. In: Dy, J., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning. Proceedings of Machine Learning Research PMLR, vol. 80, pp. 1989–1998. Stockholmsmässan, Stockholm Sweden (10–15 Jul 2018)
18. Shen, J., Qu, Y., Zhang, W., Yu, Y.: Wasserstein distance guided representation learning for domain adaptation (2017). arXiv preprint [arXiv:1707.01217](https://arxiv.org/abs/1707.01217)
19. Kumar, N., Verma, R., Sharma, S., Bhargava, S., Vahadane, A., Sethi, A.: A dataset and a technique for generalized nuclear segmentation for computational pathology. *IEEE Trans. Med. Imaging* **36**(7), 1550–1560 (2017). <https://doi.org/10.1109/TMI.2017.2677499>
20. Madani, A., Moradi, M., Karargyris, A., Syeda-Mahmood, T.: Semi-supervised learning with generative adversarial networks for chest x-ray classification with ability of data domain adaptation. In: 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), pp. 1038–1042, April 2018. <https://doi.org/10.1109/ISBI.2018.8363749>
21. Mohseni Salehi, S.S., Erdogmus, D., Gholipour, A.: Auto-context convolutional neural network (auto-net) for brain extraction in magnetic resonance imaging. *IEEE Trans. Med. Imaging* **36**(11), 2319–2330 (2017). <https://doi.org/10.1109/TMI.2017.2721362>
22. Tang, M., Perazzi, F., Djelouah, A., Ben Ayed, I., Schroers, C., Boykov, Y.: On regularized losses for weakly-supervised CNN segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 507–522 (2018)
23. Naylor, P., Laé, M., Reyat, F., Walter, T.: Nuclei segmentation in histopathology images using deep neural networks. In: 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017), pp. 933–936. IEEE (2017)
24. Kumar, N., Verma, R., Sharma, S., Bhargava, S., Vahadane, A., Sethi, A.: A dataset and a technique for generalized nuclear segmentation for computational pathology. *IEEE Trans. Med. Imaging* **36**(7), 1550–1560 (2017)
25. Naylor, P., Laé, M., Reyat, F., Walter, T.: Segmentation of nuclei in histopathology images by deep regression of the distance map. *IEEE Trans. Med. Imaging* **38**, 1–1 (2018). <https://doi.org/10.1109/TMI.2018.2865709>
26. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-CAM: visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 618–626 (2017)

27. Saito, K., Watanabe, K., Ushiku, Y., Harada, T.: Maximum classifier discrepancy for unsupervised domain adaptation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3723–3732, June 2018. <https://doi.org/10.1109/CVPR.2018.00392>
28. Hong, S., Noh, H., Han, B.: Decoupled deep neural network for semi-supervised semantic segmentation. In: Advances in Neural Information Processing Systems, pp. 1495–1503 (2015)
29. Kwak, S., Hong, S., Han, B.: Weakly supervised semantic segmentation using superpixel pooling network. In: Thirty-First AAAI Conference on Artificial Intelligence (2017)
30. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: ICML (2015)
31. Tang, M., Djelouah, A., Perazzi, F., Boykov, Y., Schroers, C.: Normalized cut loss for weakly-supervised cnn segmentation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1818–1827, June 2018. <https://doi.org/10.1109/CVPR.2018.00195>
32. Tsai, Y.H., Hung, W.C., Schuster, S., Sohn, K., Yang, M.H., Chandraker, M.: Learning to adapt structured output space for semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
33. Wollmann, T., Eijkman, C.S., Rohr, K.: Adversarial domain adaptation to improve automatic breast cancer grading in lymph nodes. In: 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), pp. 582–585, April 2018. <https://doi.org/10.1109/ISBI.2018.8363643>
34. Chen, Y.H., Chen, W.Y., Chen, Y.T., Tsai, B.C., Frank Wang, Y.C., Sun, M.: No more discrimination: cross city adaptation of road scene segmenters, pp. 2011–2020, October 2017. <https://doi.org/10.1109/ICCV.2017.220>
35. Yu, F., Koltun, V., Funkhouser, T.: Dilated residual networks. In: Computer Vision and Pattern Recognition (CVPR) (2017)
36. Zhang, Y., David, P., Gong, B.: Curriculum domain adaptation for semantic segmentation of urban scenes. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2020–2030 (2017)
37. Zhou, Y., Dou, Q., Chen, H., Qin, J., Heng, P.A.: SFCN-OPI: detection and fine-grained classification of nuclei using sibling FCN with objectness prior interaction. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)



Blood Vessel Enhancement in Liver Region from a Sequence of Angiograms Taken under Free Breathing

Morio Kawabe¹, Yuri Kokura², Takashi Ohnishi³, Kazuya Nakano³,
Hideyuki Kato⁴, Yoshihiko Ooka⁵, Tomoya Sakai⁶,
and Hideaki Haneishi³(✉)

¹ Graduate School of Science and Engineering, Chiba University, Chiba, Japan

² Graduate School of Engineering, Chiba University, Chiba, Japan

³ Center for Frontier Medical Engineering, Chiba University, Chiba, Japan
haneishi@faculty.chiba-u.jp

⁴ Department of Radiology, Chiba University Hospital, Chiba, Japan

⁵ Gastroenterological Medicine, Chiba University Hospital, Chiba, Japan

⁶ Graduate School of Engineering, Nagasaki University, Nagasaki, Japan

Abstract. Digital subtraction angiography (DSA) is one of imaging methods using X-ray image for clear visualization of the vessel information during intervention with a catheter. In order to obtain a fine DSA image, patients have to hold their breathing. However, steady breath hold is a burden for the patients and is sometimes difficult for elder patients. We propose a blood vessel enhancement method with consecutive digital angiographic images acquired under the natural breathing. Robust principal component analysis (RPCA) is used to enhance blood vessel information from consecutive angiographic images acquired under the natural breathing. RPCA can separate the consecutive images into a low-rank component and a sparse component. The information of contrast media is included in the sparse component. We implemented it on GPU and applied the proposed method to 13 sets of angiographic images and confirmed that it enables to generate satisfactory enhanced angiographic images.

Keywords: Angiographic image · Sparse model · Blood vessel image · Interventional radiology

1 Introduction

X-ray fluoroscopy is an imaging modality capable of observing internal structures and functions in real time. Spatial and temporal resolutions are better than those of other imaging modalities and interventional devices such as a catheter can also be clearly visualized. It is used to guide a catheter inserted into a blood vessel to the treated area during intervention. In this operation for liver region, contrast media is sometimes injected using a catheter to know where the current catheter top is and how the vessel pattern is located beforehand. In order to reduce the times of contrast agent injection, it is ideal to show a blood vessel road map in advance.

In this imaging, organs in irradiation direction are overlapped because X-ray fluoroscopic image is a perspective projection of 3-dimensional object to 2-dimensional image. Therefore, the background structures can be often confused with the blood vessel. In order to solve this problem, digital subtraction angiography (DSA) [1] is used. DSA is one of imaging methods using X-ray image for clear visualization of the vessel information during intervention with a catheter. DSA requires two kinds of X-ray images; mask image and live image. The mask and live images are respectively acquired before and during injection of contrast media. By subtracting the mask image from each live image, most of background structures in the live image are removed and the vessel information is enhanced. However, in the case that the target is a thoracoabdominal organ such as liver, the positions and shapes of the background structures around the vessels often change due to the physiological movement in the patient's body such as cardiac and respiratory motions. When respiratory phases of the mask and live images do not match, motion artifacts are likely to in the resultant DSA. In order to obtain a fine DSA image, patients must hold their breath. However, steady breath hold is a burden for the patients especially for elder patients.

Image registration-based methods and segmentation-based methods have been commonly proposed to remove the background structures or improve the visibility of blood vessels in X-ray fluoroscopic images. Nejati et al. proposed a registration method for cardiac angiographic images [2]. The method uses a multiscale framework in which the mask and live images are decomposed to coarse and fine sub-image blocks iteratively in order to improve the accuracy of non-rigid image registration. In [3], Xiao et al. extracted blood vessels directly from the angiographic images by automatically determining seed points and then extracting centerlines or vascular structures. A layer separation method for X-ray fluoroscopic images was proposed as another strategy for blood vessel enhancement by Zhu et al. [4, 5] and Zhang et al. [6]. This method can separate X-ray fluoroscopic images into three layers based on different motion patterns such as background structures, diaphragm, and blood vessels using multiscale framework. The visibility of blood vessels is improved, but the layer separation method also requires calculating the deformation field as these registration methods. This leads to a computational burden.

Without requiring motion estimation, robust principal component analysis (RPCA) can separate foreground moving objects from background in computer vision [7]. In [8, 9], classical RPCA was used to enhance blood vessels from complex background. Jin et al. integrate total variation (TV) regularization into the RPCA to improve enhancement of the coronary artery [10]. However, blood vessel enhancement in liver region with these techniques has not been reported to our knowledge.

In this paper, we present a method for enhancing blood vessels in liver region using consecutive angiographic images taken under free breathing. Although a general idea of the proposed method and preliminary experimental results have been presented in [11], the details of the method and experiments are described in this paper. In order to improve the reliability of the evaluation, we also significantly increased the number of datasets.

2 Methods

Figure 1 shows a processing flow of the proposed method. The pixel value of each frame of the angiographic motion picture is first logarithmic-transformed. The X-ray intensity distribution detected by the image sensor would obey the Lambert-Beer's law and modeled as

$$I(x, y, t) = I_0 \exp[-\mu_{org}(x, y, t) - \mu_{con}(x, y, t)]. \quad (1)$$

Here I_0 is the incident radiation. $\mu_{org}(x, y, t)$ represents the integration of attenuation coefficient of organs along the line connecting the X-ray source and the detected position (x, y) and $\mu_{con}(x, y, t)$ represents that of the contrast agent. Two attenuation terms exist in the exponential function. By normalizing the detected image by the incident radiation and taking its logarithm, two components are modeled by a linear sum as

$$g(x, y, t) = -\log(I(x, y, t)/I_0) = \mu_{org}(x, y, t) + \mu_{con}(x, y, t). \quad (2)$$

Image $g(x, y, t)$ is more suitable than $I(x, y, t)$ for the purpose of separating organ image and contrast agent image. After such preprocessing, RPCA-based component separation is performed. Contrast agent is separated as a sparse component after exponential transformation is applied. Exponential transformation is used as the inverse of logarithmic transformation. Details of RPCA are given in the following sub-sections.

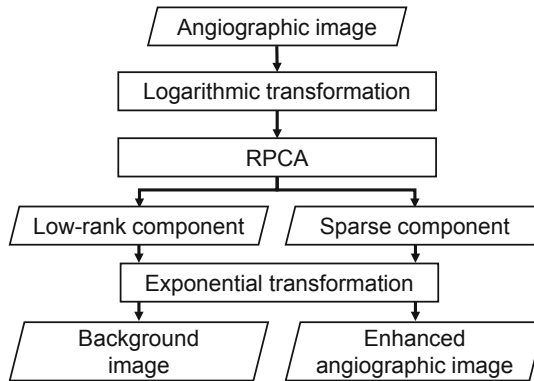


Fig. 1. Processing flow of the blood vessel enhancement method

2.1 Blood Vessel Enhancement

Classical RPCA is used to enhance blood vessel information without any omission. RPCA can separate the consecutive angiographic images Γ into a low-rank component \mathbf{L} and a sparse component \mathbf{S} . Here, each column of Γ is composed of the pixel values of a certain time frame and each row is composed of time sequential pixel values at a

certain position. This decomposition is formulated as the following optimization problem:

$$\min\{\|\mathbf{L}\|_* + \lambda\|\mathbf{S}\|_1\} \text{ subject to } \mathbf{\Gamma} = \mathbf{L} + \mathbf{S} \quad (3)$$

where $\|\cdot\|_*$ denotes the nuclear norm of a matrix, i.e., the sum of singular values of the matrix and $\|\cdot\|_1$ denotes the l_1 -norm, i.e., the sum of the absolute values of matrix entries. λ is the trade-off parameter to strike a balance between the two norms. If λ is a high value, a lot of information is classified into the low-rank component while the sparse component is almost empty. The low-rank component represents background structures and periodical information such as bones and organ motion along with breathing. The sparse component represents rapid and non-periodic information such as the flow of the contrast media and the motion of the high-contrast catheter.

Although the conventional RPCA works for above-mentioned separation to some extent, there are at least two defects. (1) Artifacts due to complicated motion caused by intestinal gas. Spatially complicated pattern and temporally fast motion of intestinal gas are likely to be classified into the sparse component. So intestinal gas may yield false vessel pattern. (2) The liver region around diaphragm has significant and rapid change in pixel values due to respiration and likely to be classified into the sparse component. Therefore, we introduced further constraints based on two types of a-priori knowledge:

Knowledge 1: Pixel values in the same organ are similar and sharp changes in pixel values take place around the border of organs. Blood vessels have these characteristics and are also continuous in space and time.

Knowledge 2: Blood vessels with contrast agent have smaller pixel values (dark) than surrounding area (intensity level).

Based on Knowledge 1, we introduced a term of total variation (TV) penalty. Since TV penalty intends to maintain smooth intensity distribution and to remove isolated dots or small regions, it is expected to remove artifacts caused by intestinal gas as keeping continuous blood vessels as they are. However, since the liver region around diaphragm has a smooth and large structure, TV penalty does not work. Thus, we introduced the second penalty, which works as a restriction on the range of pixel value. In general, X-ray absorption by blood vessels with contrast media is higher than liver and other soft tissue. In our implementation, we empirically set the condition for pixel values of the sparse component so that it is positive.

This decomposition is formulated as the following optimization problem:

$$\min\left\{\|\mathbf{L}\|_* + \lambda_S\|\mathbf{S}\|_1 + \lambda_{TV}\|\nabla\mathbf{T}\|_{2,1}\right\} \text{ subject to } \mathbf{\Gamma} = \mathbf{L} + \mathbf{S}, \mathbf{S} > 0, \mathbf{T} = \mathbf{S} \quad (4)$$

$$\|\nabla\mathbf{T}\|_{2,1} = \sum_i \sqrt{(\nabla_x\mathbf{T})_i^2 + (\nabla_y\mathbf{T})_i^2 + (\nabla_t\mathbf{T})_i^2} \quad (5)$$

where $\|\cdot\|_{2,1}$ denotes the $l_{2,1}$ -norm. λ_S , and λ_{TV} are the control parameter regarding sparse component and TV regularization, respectively. Equation (5) reduces intermittent noise appearing in the blood vessel image while preserving edges by minimizing

the first derivative of the x , y , and time axes. Equations (3) and (4) are minimized by using the alternating direction method of multipliers (ADMM) [12, 13]. In updating total variation term, the Split Bregman method was used [14].

The termination condition in the optimization of Eq. (4) is formulated as follows:

$$\|\mathbf{L}_{j+1} - \mathbf{L}_j\|_2 + \|\mathbf{S}_{j+1} - \mathbf{S}_j\|_2 < tol \quad (6)$$

where j is the number of iterations, and tol is the convergence tolerance. tol was empirically set to 0.001.

2.2 Parallel Processing

For fast optimization, we implemented GPU parallel processing. Figure 2 shows the processing flow of parallel processing. First, angiography dataset is transferred from CPU to GPU. Subsequently, the TV regularization image is updated using Split Bregman method. Then the background component and the blood vessel component are updated. After updating the components, the convergence is judged by the termination condition. To save transfer time, only scalar values were transferred from GPU to CPU. The GPU side is instructed to output each component only if the termination condition is satisfied.

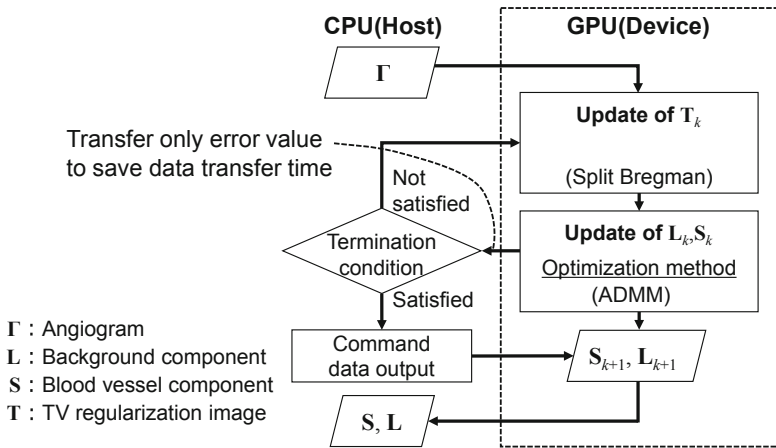


Fig. 2. Processing relation between CPU and GPU

3 Experiments

Image acquisition experiments were conducted with 13 patients. This study was approved by the Ethical Review Board of Chiba University and all patients gave informed consent to participate in this study. Contrast enhancement methods were applied to free breathing angiograms. The target site is the celiac artery, and the image size is around 900×700 pixels, the number of frames is around 50–60 frames. In the used computer, CPU was Core i7-6850 K (Intel) with 6 cores and 128 GB RAM and

GPU was GeForce GTX1080 (NVIDIA) with 2560 cores and 8 GB RAM. In order to accelerate all methods, Compute Unified Device Architecture was introduced.

As for the trade-off parameter λ in the formulation of conventional RPCA, we set it to $\lambda = 0.6/\sqrt{N}$ and modified RPCA integrated with TV regularization uses $\lambda_S = 0.7/\sqrt{N}$, $\lambda_{TV} = 0.3/\sqrt{N}$, where N is the number of pixels of time-sequential 2-dimensional image. λ uses the same value as [7]. λ_S, λ_{TV} were determined after grid search over wide ranges. These three parameters were used for all datasets.

The performance of the proposed method was qualitatively and quantitatively evaluated through comparing with the other two methods. One is ‘‘DSA’’ in which background structure is eliminated by subtracting a mask image which is given by the median of the first five frames in the sequence. The other method is the conventional RPCA using Eq. (3), named ‘‘Conventional RPCA’’.

Corrected contrast-to-noise ratio (cCNR) proposed by Ma et al. [15] is used as the method of quantitatively evaluation. Once the foreground and background of an image are defined, the definition of cCNR can be formulated as:

$$cCNR = \frac{|\mu_F - \mu_B|}{\sqrt{\sigma_B^2 + k \cdot MSE^V}} \quad (7)$$

$$MSE^V = \sum_{x,y} (I_{\text{truth}}^V(x,y) - I_S^V(x,y))^2 / |I_{\text{truth}}^V| \quad (8)$$

where μ_F and μ_B are the mean of foreground and background pixel values respectively, σ_B is the standard deviation of the background pixel values, and k is a weighting factor that strike a balance between σ_B and MSE^V . k was empirically set to 1/10. Equation (8) evaluates the degree of defect in the blood vessel region based on the difference from the ideal blood vessel image. An ideal DSA is used for an ideal blood vessel image. It was created by using an ideal mask image which has the smallest dispersion with the live image. cCNR measures the contrast between the foreground and background pixel intensities in relation to the standard deviation of the background pixel intensities. Larger cCNR values imply a better contrast. Figure 3 shows an example of foreground and background regions. Background is defined as the white image region and foreground being the dark area within the white part. These regions (binary mask) were manually determined. In the experiment, we select 5 frames from each sequence for the mask generation and compute the average cCNR of the 5 frames.

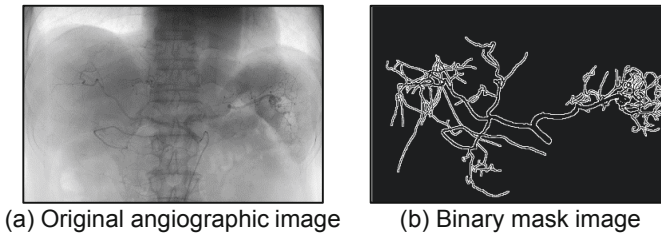


Fig. 3. Definition of foreground and background for calculation of cCNR

4 Results and Discussion

4.1 Results

Figure 4 shows an example of comparison of the enhancement methods. Here we selected the same timing frame from a sequence of images. At this timing the contrast medium was widely flowed to the organ. The result of “DSA” shows strong artifacts because the background structures between the mask image and the live images are different by respiratory motion and heartbeat. “Conventional RPCA” enhanced angiographic images without requiring the mask images, but still presents artifacts around intestines with gas and diaphragm region. On the contrary, “Modified RPCA” reduced artifacts and improved the visibility of blood vessels.

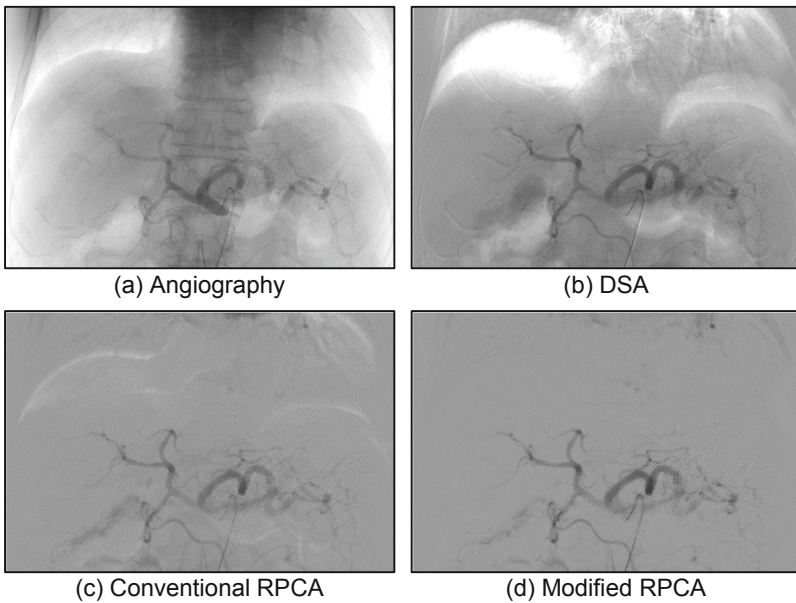


Fig. 4. Comparison of blood vessel enhancement results

Figure 5 shows the cCNR values. The cCNR values of “modified RPCA” were higher than those of other methods. It means that the visibility of blood vessels was improved.

The total processing time after acquisition of a sequence of angiograms was 7.9 ± 0.5 [s]. Our co-authors include medical doctor, and he suggested the process be completed in 20–30 s. Although only one medical doctor’s evaluation is insufficient for clinical practice, we suppose that it is short enough.

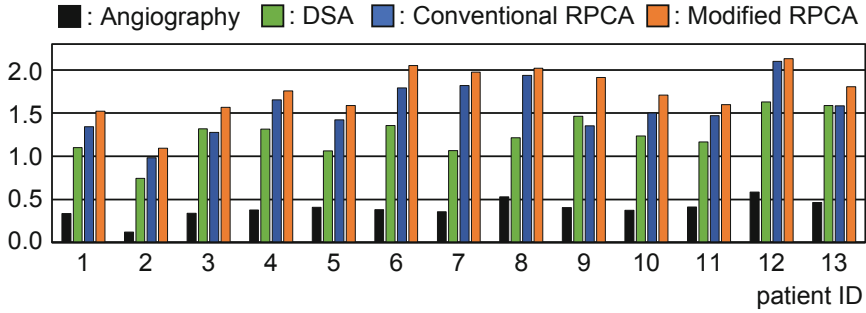


Fig. 5. Values of calculated cCNR for each patient dataset

4.2 Discussion

As mentioned in introduction, the final goal of this work is to make a blood vessel pattern image which can work as a blood vessel road map before catheter guidance. However, injected contrast media does not necessarily show clear and continuous vessel pattern. The obtained blood vessels in each frame sometimes have gap caused by body motion, such as breathing or intestinal motion. In this paper, we presented a method for blood vessel enhancement and applied it to each frame of angiogram. However, to generate a complete road map of blood vessels, we need another step to provide those images. One solution might be registration and integration of those images. This is one of our future works.

While the large or moderate size of blood vessels are successfully classified into sparse component and enhanced for visualization, very thin vessels may disappear. However, this is not a serious drawback because the visualization of such thin vessels is not always needed in the generation of blood vessel road map for catheter guidance.

5 Conclusions

In this paper, we presented a blood vessel enhancement method using consecutive angiographic images with respiratory motion. The proposed method was based on robust principal component analysis to separate the original motion picture into low-rank component and sparse component. We added penalty terms to the sparse component for better performance of blood vessel extraction. In order to achieve high-speed processing, we implemented parallel processing using GPU. We applied the proposed method to 13 patient datasets. In all cases we confirmed both qualitatively and quantitatively that visibility of blood vessels was improved by the proposed method. Using the parallel processing, total processing was successfully completed in about 8 s, which is short enough for clinical use in practice.


Acknowledgment. This work was supported in part JSPS KAKENHI (16K16406) and the JSPS Core-to-Core Program.

References

1. Kruger, A.R., et al.: Computerized fluoroscopy in real time for noninvasive visualization of the cardiovascular system. *Radiology* **130**, 49–57 (1979)
2. Nejati, M., Hossein, P.: Multiresolution image registration in digital X-ray angiography with intensity variation modeling. *J. Med. Syst.* **38**, 1–10 (2014)
3. Nejati, M., Hossein, P.: Multiresolution search strategy for elastic registration of X-ray angiography images. In: *Intelligent Computation and Bio-medical Instrumentation (ICBMI)*, pp. 216–219 (2011)
4. Zhu, Y., Prummer, S., Chen, T., Ostermeier, M., Comaniciu, D.: Coronary DSA: enhancing coronary tree visibility through discriminative learning and robust motion estimation, In *Proc. of SPIE 7259* (2009)
5. Zhu, Y., Prummer, S., Wang, P., Chen, T., Comaniciu, D., Ostermeier, M.: Dynamic layer separation for coronary DSA and enhancement in fluoroscopic sequences. In: Yang, G.-Z., Hawkes, D., Rueckert, D., Noble, A., Taylor, C. (eds.) *MICCAI 2009*. LNCS, vol. 5762, pp. 877–884. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04271-3_106
6. Zhang, W., Ling, H., Prummer, S., Zhou, K.S., Ostermeier, M., Comaniciu, D.: Coronary tree extraction using motion layer separation. In: Yang, G.-Z., Hawkes, D., Rueckert, D., Noble, A., Taylor, C. (eds.) *MICCAI 2009*. LNCS, vol. 5761, pp. 116–123. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04268-3_15
7. Candès, E.J., Li, X., Ma, Y., Wright, J.: Robust principal component analysis? *J. ACM* **58**, 1–37 (2011)
8. Brosig, R., Hariharan, S.G., Volpi, D., Kowarschik, M., Carlier, S., Navab, N., et al.: Implicit background subtraction for cardiac digital angiography. In: *Joint MICCAI Workshops on Computing and Visualization for Intravascular Imaging and Computer-Assisted Stenting*, pp. 50–57 (2015)
9. Ma, H., et al.: Layer separation for vessel enhancement in interventional X-ray angiograms using morphological filtering and robust PCA. In: Linte, C.A., Yaniv, Z., Fallavollita, P. (eds.) *AE-CAI 2015*. LNCS, vol. 9365, pp. 104–113. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24601-7_11
10. Jin, M., Li, R., Jiang, J., Qin, B.: Extracting contrast-filled vessels in X-ray angiography by graduated RPCA with motion coherency constraint. *Pattern Recogn.* **63**, 653–666 (2017)
11. Kawabe, M., et al.: GPU-accelerated blood vessel enhancement from free-breathing angiography using robust principal component analysis. *J. Comput. Assist. Radiol. Surg.* **14** (Supplement No. 1), S12 (2019)
12. Gabay, D., Mercier, B.: A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Comput. Math Appl.* **2**(1), 17–40 (1976)
13. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.* **3**(1), 1–122 (2011)
14. Goldstein, T., Osher, S.: The split Bregman method for L1-regularized problems. *SIAM J. Imaging Sci.* **2**(2), 323–343 (2009)
15. Ma, H., Hoogendoorn, A., Regar, E., Niessen, W.J., van Walsum, T.: Automatic online layer separation for vessel enhancement in X-ray angiograms for percutaneous coronary interventions. *Med. Image Anal.* **39**, 145–161 (2017)



Real-Time Morphing of the Visible Man Liver with Intrahepatic Vasculatures

Maxime Berg¹, Changwei Zhang², and Harvey Ho³(✉) 

¹ Department of Fluid Mechanics and Hydraulics, ENSEEIHT, National Polytechnic Institute of Toulouse, Toulouse, France

² Neurosurgery Department, West China Hospital, Chengdu, Sichuan, China

³ Auckland Bioengineering Institute, The University of Auckland, Auckland, New Zealand
harvey.ho@auckland.ac.nz

Abstract. Computational models for liver deformation are usually performed without considering intrahepatic vasculatures. The major hurdle is the computational cost when deforming the liver and its vessels simultaneously. In this paper we introduce a numerical method containing a combined constrained constructive optimisation (CCO) algorithm and host mesh fitting (HMF) algorithm. While the CCO algorithm is used to generate a large liver vascular network, the HMF algorithm morphs hepatic structure within a host mesh. This technique is applied to the liver of the Visible Man (VM), where total 16,300 vessels are generated to extend the 84 digitised portal and hepatic veins in the VM liver. The liver deformation due to respiration effects and heart beats is simulated in real-time (35 Hz) and matched with the video sequence of an endovascular Trans-Arterial Chemo Embolization (TACE) procedure. In conclusion an efficient method for morphing a virtual liver containing large vasculatures is proposed, and may have applications in chemotherapy and endovascular simulations.

Keywords: Liver · Deformable model · Visible Man · Host mesh fitting

1 Introduction

The liver is heavily vascularized to serve its roles in the filtration and storage of blood [1]. Being a soft organ, the liver is under constant shape changes due to heart beats and respiration effects. Hence an ideal liver model would have the liver parenchyma and intra-hepatic structures deformed simultaneously. However, computer algorithms for liver deformation usually do not deal with the concurrent distortion of liver and its intra-hepatic vasculatures (for a review see [2]). This problem has been addressed in a few recent works, e.g. in a subject-specific liver model including the parenchyma, the Glisson's capsule and blood

vessels, where the hepatic structures are deformable and presented in the context of laparoscopy surgery [3]. Specifically, the vascular tree in [3] was based on the skeletonisation of segmented blood vessels from CT images, and the vascular tree was constructed as beam elements suitable for finite-element analysis (FEA). This approach, however, can only handle a small number of vessels, e.g., tens of vessels in the model. To apply this method to a large vasculature containing hundreds or thousands of vessels, a different approach is required to circumvent the computational bottleneck.

Two challenges need to be addressed here when deforming a virtual liver and its vascular trees. Firstly a suitable geometric representation for the vascular tree is required. A 3D vascular model demands a high cost in graphical rendering and deformation, and becomes computationally prohibitive with a large vasculature. It is also not essential to visualise all blood vessels in their 3D details, and indeed a majority of them are not detectable from clinical medical images, where the image resolution may reach 0.5 mm. In this work we use a constrained constructive optimisation (CCO) algorithm to generate 1D vascular trees. The algorithm was originally proposed in [4] for a 2D circular tissue, and has been used to generate large hepatic vasculatures for structural analysis [5], hepatic clearance [6] and hepatic arterial flow modelling [7]. In [7], the first several generations of arteries were digitised from CT images at first, and the CCO algorithm was used for creating small vessels downstream the larger vessels. We will adopt the same method for the hepatic tree generation.

Secondly the numerical algorithm for simultaneously deforming the liver and its vessels is the key for computational cost. A FEA analysis for the liver parenchyma needs to handle complex constitutive equations governing soft tissue deformation, even without considering embedded vessels. If validated such an analysis would again be too computationally expensive. However, since the hepatic motion due to respiration effects is small (1–2.6 cm at the cranio-caudal direction [8]), we can take advantage of the fact that the topology of the vasculature remains unchanged while being reformed, and use a so-called host-mesh-fitting (HMF) algorithm. This is a Finite Element mesh-based geometric modelling method and has been used to simulate the displacement of skeleton muscles [9] and the heart [10]. A similar workflow can be applied to the liver and its vessels.

The aim of this work is to combine the CCO and the HMF algorithm for real-time simulations of a virtual liver containing large vasculatures. This method may be used for modelling hepatic motions in an endovascular procedure such as the Trans-Arterial Chemo Embolization (TACE) for treating hepatocellular carcinoma tumours.

2 Methods

2.1 Digitisation of Blood Vessels from the Visible Man

We make use of the Visible Man (VM) data set from the National Library of Medicine (NLM, Bethesda, MD) to exemplify the algorithm. The data set of VM

contains 2D images slices of $2,048 \times 1,216$ pixels, and is available from public domains [15]. The intra-slice and interslice resolutions are 0.33 mm and 1 mm, respectively. The same VM data set has been used to illustrate the concept of virtual reality in liver surgery [14] and to describe the liver anatomy [16]. In [14] the liver surface was extracted by using a semi-automatic deformable model, and 14,000 triangles were used to represent the segmented liver surface.

Our method differs from that of [14] and [16] in that a parametric cubic Hermite mesh is used to represent the various hepatic structures, i.e. a 1D mesh for the vascular tree, a 2D surface mesh for the liver surface or Glisson's capsule, and a 3D volume mesh for the liver parenchyma. The process is illustrated in Fig. 1. In Fig. 1(a), key points (nodes) were manually placed along the contours of the VM liver, then a bicubic Hermite mesh was constructed (Fig. 1b). In Fig. 1(c), cylinders were used to represent blood vessels, where nodes were placed along the centreline and radius data recorded for each node. Using this method eight generations total 84 portal venous (PV) and hepatic venous (HV) vessels are digitised (Fig. 1c). The final mesh contains a combination of 1D, 2D and 3D elements, as shown in Fig. 1(d).

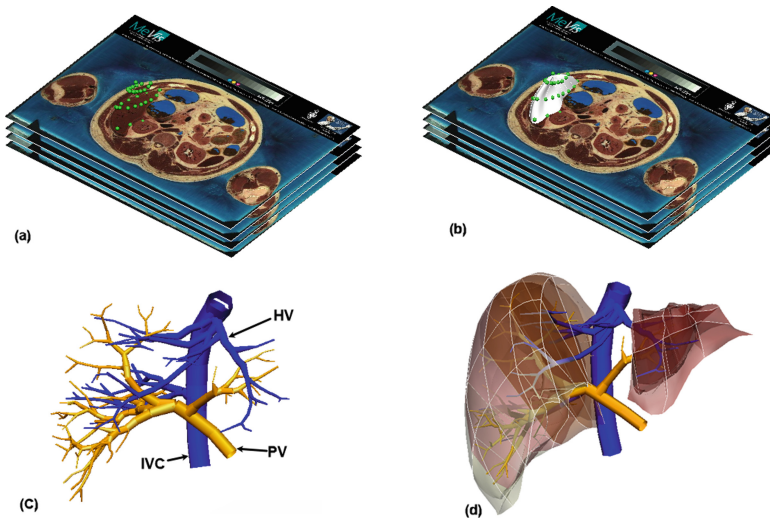


Fig. 1. Digitised Visible Man liver: (a) a data cloud digitised at the contours of the liver; (b) a bicubic Hermite mesh is constructed based on the data cloud; (c) PV and HV trees of up to eight generations are digitised; (d) an overall impression of the VM liver.

The PV and HV trees of the VM shown in Fig. 1 represent the typical liver vessel branching pattern [16]. The diameter of the smallest vessel in the digitised hepatic tree is about 1.2 mm. The digitisation is a manual process and takes up to four hours to construct in the open source software Cmgui (<https://www.cmiss.org/cmgui>). The hepatic arterial tree was not constructed

because it was barely visible in the VM images due to lose of blood pressure post-mortem.

2.2 An Implementation of the CCO Algorithm

In the CCO algorithm, the growth of small blood vessels follows the principle of minimum energy, i.e., the vascular network uses a minimum energy to perfuse a tissue [4]. The core of this method consists of the minimization of a target function, which is the bifurcation volume [4]:

$$V = \pi \sum_{i=1}^N r_i^2 l_i \quad (1)$$

where $N = 3$ represents the three vessel segments in a bifurcation, V is the total blood volume, r and l are the radius and length of a vessel segment, respectively. The blood flow in the tree is approximated as Poiseuille flow and satisfies the relationship between pressure drop ΔP , resistance and flow rate Q :

$$\Delta P = \frac{8l\mu}{\pi r^4} Q \quad (2)$$

where μ is the viscosity of the blood. For the PV tree the flow rate at the root portal vein was set as 900 mL/min or 15 mL/s, and the pressure at the terminals of PV tree as 9 mmHg [12]. The flow in the HV tree was created in a reversed manner, i.e. the outlets of the HV were virtually viewed as inlets and the perfusion rate was 1,350 mL/min. The pressure drop across the HV tree was 2 mmHg, which means that the resistance to the hepatic blood returning to the heart is very small [12].

The graphical realism of a vasculature depends on the global or structural criterion, and local or geometric criterion. The former determines the shape of the tree, and the latter defines the shape of bifurcations. Concerning the radius of parent and daughter vessels, the power law branching pattern was used [13]:

$$r_0^\lambda = r_1^\lambda + r_2^\lambda \quad (3)$$

where 0, 1, 2 represent the parent vessel and two daughter vessels. λ is a constant and is configured as 2.7 in our implementation, in accordance with the suggested value between 2 to 3 in [5]. The following procedure is followed to add a new vessel segment to a current tree (Fig. 2):

1. Randomly generate a point within the perfusion volume;
2. Search for the closest segments to the point;
3. Perform the optimization process to create a bifurcation;
4. Check if every constraint is satisfied;
5. Generate a list of candidates which passed Step 4;
6. Use the one candidate which have the smallest tree volume

In Step (3), minimization of the target function is performed using a trust-region based algorithm [17], which is suitable for convex shape problems. In Step (4) geometric constraints are enforced so that the bifurcation angles and the length of each new vessel are controlled within physiological values [13]. In addition, all vessels must be contained inside the perfusion volume and do not intersect with any other vessels from the same tree [5].

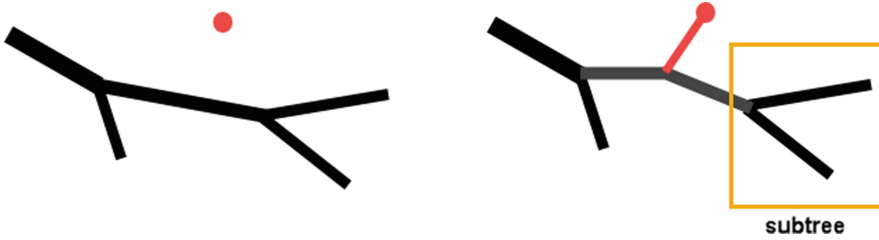


Fig. 2. Adding a new branch in the tree: after a random point is added into the perfusion volume, the algorithm will search for the closest segment to it, and an optimisation routine is run to determine the bifurcation point that gives the minimum tree volume.

2.3 Host Mesh Fitting Algorithm

The HMF algorithm may be considered as one of the free-form deformation techniques used in computer graphics to manipulate 3D objects [9]. The transformation is an affine transformation as it preserves collinearity and ratios of distances. The HMF algorithm involves the use of two parametric meshes, i.e. a host mesh and a slave mesh (Fig. 3), where the slave mesh contains the objects of interest, in this case the liver and its blood vessels. The host mesh is an assistive mesh that encapsulates the slave mesh. Figure 3 shows a diagram of the host and the slave mesh with their coordinate systems (η and ξ). The nodal coordinates of the slave mesh can be expressed as:

$$\eta_i = f_i(\xi_1, \xi_2, \xi_3) \quad (4)$$

where $i = 1, 2, 3$, f_i is the parametric function of the nodal coordinates of the host mesh, which is the same as the basis functions (e.g. linear Lagrange or cubic

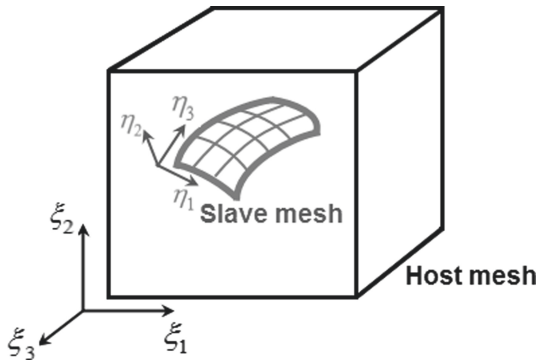


Fig. 3. Diagram of the host and slave mesh. The slave mesh (η_i) is contained within the host mesh (ξ_i), and the relative nodal position of the slave mesh in the host mesh remain unchanged when the deformation is small.

Hermite functions) used in a finite element mesh [9]. When the deformation of liver is small and the topology of the liver mesh is consistent, i.e. no tearing or cutting of the mesh occurs, f_i can be approximated as unchanged. This implies that the relative locations of the nodes on the slave mesh with respect to the host mesh remain identical. When the host mesh is deformed, the displacements of the host mesh drive the morphing of the slave mesh, and dynamically update its nodal positions.

3 Results

3.1 Creation of Large Vasculatures for the VM Liver

The CCO algorithm was applied to the digitised PV and HV trees shown in Fig. 1. From the existing 84 vessel digitised, small veins and venules were generated. Overall thirteen generations of total 16,384 blood vessels were created for the PV and HV trees. The flow rate and pressure drop in each of the blood vessels were solved according to Eq. (2). It can be seen from Fig. 4 that the geometric constraints of the algorithm are obeyed as all newly generated vessels are contained within the liver organ.

3.2 Morphing of the Liver and Its Vasculatures

The multi-dimensional mesh shown in Fig. 4, including the liver surface and vasculatures, are treated as a single slave mesh and placed in the host mesh of a trilinear element of eight nodes (Fig. 5). Since it was infeasible to obtain body motion information from the VM dataset because the images were obtained post-mortem, we utilised the data from two sources to help with the simulation. Firstly, we had the hepatic motion data as reviewed in [8]. Secondly, the fluoroscopy video of a TACE procedure for the treatment of an unresectable liver tumour was used as a visual guidance. The video, shown in several snapshots in Fig. 5, illustrates the parenchymal displacements. In the left column of Fig. 5, four video frames with 1 second apart (denoted +1s, +2s, +3s and +4s) are shown. The white triangle indicates the position of a guide wire for the catheter. In the right column, the deformation of the virtual liver is driven by the host mesh, corresponding to the TACE video.

It is clear from the review in [8] that the largest liver displacement (1–2.6 cm) occurs at the cranio-caudal direction, while the motions in other directions are insignificant. This is confirmed from the video sequence of the TACE procedure. Moreover, it can be seen from the TACE video that the inferior tip of the liver is unaffected from the respiration effects. Based on these observations, the upper four nodes of the host cubic mesh were programmed to displace at the cranio-caudal direction with the total displacements of 2.5 cm in each motion cycle. Motion vectors of the four nodes are shown in Fig. 5, which in turn drove the motion of the liver and its vasculatures. The simulation was run on a desktop computer (Intel Core i5-4690 CPU @ 3.5 GHz, RAM 16 GB). The responding time for each nodal displacement iteration was 0.028 s, or 35 Hz.

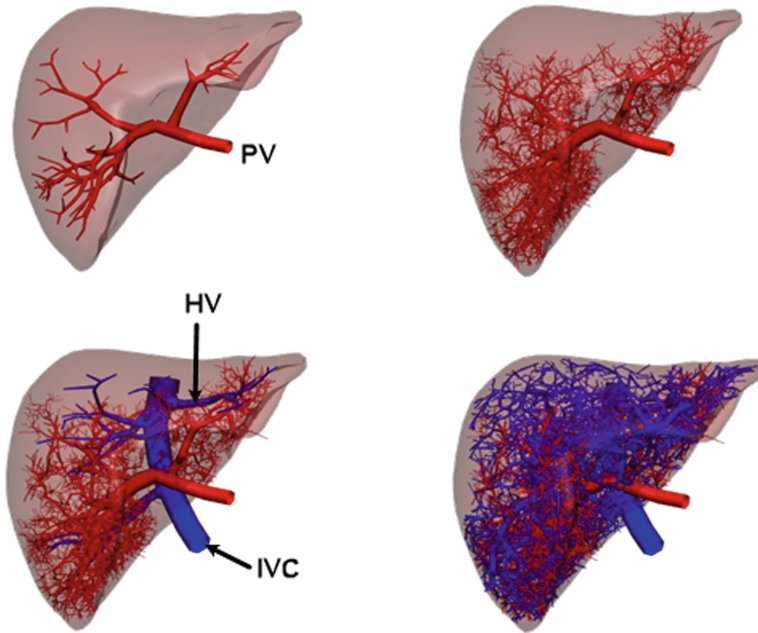


Fig. 4. PV and HV vasculatures for the liver. Top left: the digitised PV tree from the VM dataset; Top right: the PV tree was extended to thirteen generations using the CCO algorithm; Bottom left: inclusion of the HV tree plus IVC (in blue colour); Bottom right: the final expanded PV and HV trees with total 16,300 vessels. (Color figure online)

4 Discussion

In silico models are valuable in many clinical and biomedical applications such as surgical planning, image-based navigation, chemotherapy and drug effect simulations. In drug delivery applications, it is desirable to extend a vascular tree from a spatial level (~ 1 mm) visible from clinical CT/MRI images to a sub-millimeter level where the drug agent is released. For microsphere-carried-drug delivery, a vascular tree detailed to the arteriole or venule level, or even to the sinusoidal or cellular level would be useful, because that is the spatial level where the uptake of drug by hepatocytes occurs. Vascular morphology information at such a fine resolution cannot be achieved from clinical CT/MRI imaging, but may be feasible with other imaging modalities such as micro-CT [11], intravital microscopy [18], etc. Hence, the construction of a large vasculature needs to connect the vascular models from different spatial scales, and from different imaging modalities. The CCO algorithm demonstrated in the paper is capable of generating a vascular network spanning from the scale of millimetres, i.e. the site of drug releasing, to the scale of micrometres of capillaries. This provides an attractive solution to the simulation of TACE where the arterial supply to a tumour and hepatic clearance need to be considered from multiple spatial scales [6].

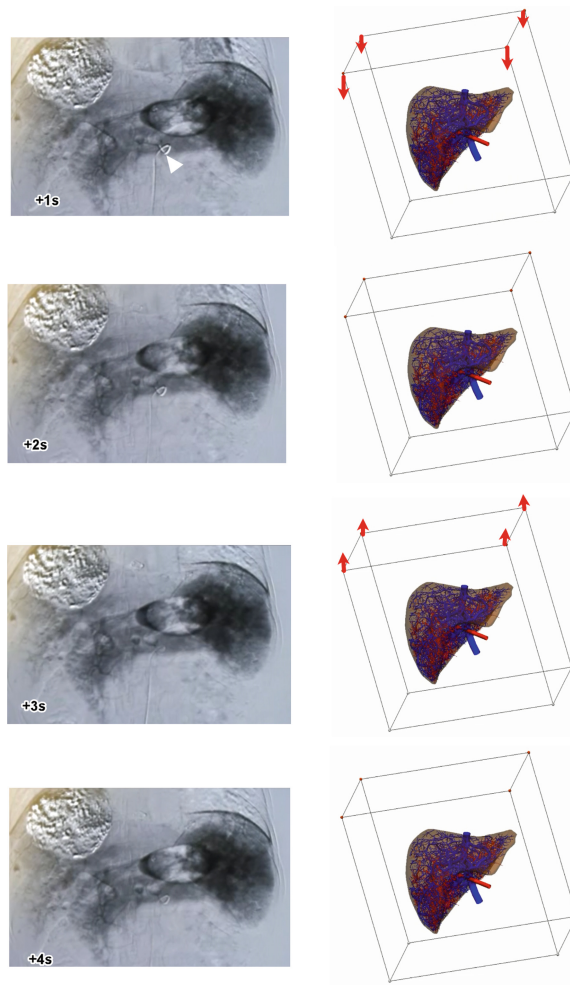


Fig. 5. Simulation of the liver motion. Left column: snapshots of the fluoroscopy video of a TACE procedure. Right column: the motion of the virtual liver corresponds to the video sequence. The vectors indicate nodal displacements of the host mesh, which are 2.5 cm at the cranio-caudal direction.

The major challenge we intended to address in this work was to deform a large vasculature in a time frame relevant to clinical navigation applications. As shown in Fig. 5, the simulation of the liver motion following diaphragm displacements was run in real-time. The assumption made for the simulation was that the relative locations of intra-hepatic structures were identical under small deformation, thus spared the cost of expensive Finite Element analysis. The time saved can be used in tracking extra-hepatic fiducial markers, or the nodes of the host mesh. The markers may be placed at the surface of the abdomen, which can

be detected by a vision camera in real-time [19]. Since the host mesh has much less nodes (in the case of Fig. 5 it is 8) than the slave mesh, the computational cost for their tracking is more affordable. Moreover, the number of elements in the host mesh is much smaller than that of the slave mesh (in the case of Fig. 3 there is only one element in the host mesh). Therefore, it is much more computationally efficient to use the resultant transformation matrix to deform the slave mesh.

There are some limitations pertaining to the current model. While we have used a computer-generated vascular structure model, which was built upon a healthy liver of the VM liver, to simulate the deformation of a diseased liver (as in Fig. 5), we did not consider the mechanical and physiological differences between them. For liver diseases such as cirrhosis, nonalcoholic steato-hepatitis, etc, the density of liver parenchyma may be different, and hence affect the liver's mechanic properties [20]. Moreover, the generation of small vessels through CCO is a random process. Currently there is no correlation with the data from the patient, except the coarse initial vessels. And a clinical validation of the generation process in terms of density of vessels, average orientations, etc have not been performed.

Nevertheless, the computational technique presented in the paper is the first of its kind that deforms a large vasculature in the order of 10,000 vessels in real-time, to our knowledge. The proposed approach can have applications where a detailed vascular tree and its deformation are important, for example in treating hepato-cellular carcinoma.

5 Conclusion

In this paper we presented a computational technique based on a combined CCO and HMF approach to deform a virtual liver and its vasculature in real-time. The technique may have potential applications in surgical navigation and chemotherapy for the liver.

References

1. Hall, J.E.: *Guyton and Hall Textbook of Medical Physiology*. Elsevier, Amsterdam (2015)
2. Meier, U., López, O., Monserrat, C., Juan, M.C., Alcañiz, M.: Real-time deformable models for surgery simulation: a survey. *Comput. Methods Programs Biomed.* **77**, 183–197 (2005)
3. Plantefève, R., Peterlik, I., Haouchine, N., Cotin, S.: Patient-specific biomechanical modeling for guidance during minimally-invasive hepatic surgery. *Ann. Biomed. Eng.* **44**, 139–153 (2016)
4. Schreiner, W., Buxbaum, P.F.: Computer-optimization of vascular trees. *IEEE Trans. Biomed. Eng.* **40**, 482–491 (1993)
5. Schwen, L.O., Preusser, T.: Analysis and algorithmic generation of hepatic vascular systems. *Int. J. Hepatol.* **2012**, e357687 (2012)

6. White, D., Coombe, D., Rezania, V., Tuszynski, J.: Building a 3D virtual liver: methods for simulating blood flow and hepatic clearance on 3D structures. *PLoS One* **11**, e0162215 (2016)
7. Muller, A., Clarke, R., Ho, H.: Fast blood-flow simulation for large arterial trees containing thousands of vessels. *Comput. Methods Biomech. Biomed. Eng.* **20**, 160–170 (2017)
8. Clifford, M.A., Banovac, F., Levy, E., Cleary, K.: Assessment of hepatic motion secondary to respiration for computer assisted interventions. *Comput. Aided Surg.* **7**, 291–299 (2002)
9. Fernandez, J.W., Mithraratne, P., Thrupp, S.F., Tawhai, M.H., Hunter, P.J.: Anatomically based geometric modelling of the musculo-skeletal system and other organs. *Biomech. Model. Mechanobiol.* **2**, 139–155 (2004)
10. Wang, V.Y., Lam, H.I., Ennis, D.B., Cowan, B.R., Young, A.A., Nash, M.P.: Modelling passive diastolic mechanics with quantitative MRI of cardiac structure and function. *Med. Image Anal.* **13**, 773–784 (2009)
11. Nordsletten, D.A.: Structural morphology of renal vasculature. *AJP: Heart Circul. Physiol.* **291**, H296–H309 (2006)
12. Ho, H., Sorrell, K., Bartlett, A., Hunter, P.: Modeling the hepatic arterial buffer response in the liver. *Med. Eng. Phys.* **35**, 1053–1058 (2013)
13. Zamir, M.: On fractal properties of arterial trees. *J. Theor. Biol.* **197**, 517–526 (1999)
14. Marescaux, J., Rubino, F., Arenas, M., Mutter, D., Soler, L.: Augmented-reality-assisted laparoscopic adrenalectomy. *JAMA* **292**, 2214–2215 (2004)
15. Ackerman, M.J.: The visible human project: a resource for education. *Acad. Med.* **74**(6), 667–670 (1999)
16. Fasel, J.H., et al.: Liver of the “visible man”. *Clin. Anat.* **10**, 389–393 (1997)
17. Moré, J., Sorensen, D.: Computing a trust region step. *SIAM J. Sci. Stat. Comput.* **4**, 553–572 (1983)
18. Meyer, K., et al.: A predictive 3D multi-scale model of biliary fluid dynamics in the liver lobule. *Cell Syst.* **4**, 277–290.e9 (2017)
19. Yu, H.B., Ho, H.: System designs for augmented reality based ablation probe tracking. In: Paul, M., Hitoshi, C., Huang, Q. (eds.) *PSIVT 2017. LNCS*, vol. 10749, pp. 87–99. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-75786-5_8
20. Angulo, P.: Nonalcoholic fatty liver disease. *N. Engl. J. Med.* **346**(16), 1221–1231 (2002)



Development of 3D Physiological Simulation and Education Software for Pregnant Women

Aurélien Bourgeois¹, Sarah Ancé², and Harvey Ho³(✉) 

¹ Arts et Métiers ParisTech, Paris, France

² IMT Lille-Douai, Douai, France

³ Auckland Bioengineering Institute,

The University of Auckland, Auckland, New Zealand

harvey.ho@auckland.ac.nz

Abstract. Many women smoke during pregnancy despite the harmful effects of maternal smoking on the fetus being well established. Providing women with better support to stop smoking when pregnant is likely to increase their motivation to quit. We aim to combine 3D modelling and physiological simulations to create a science-based educational tool for pregnant women. We employ parametric mesh (linear Lagrange or cubic Hermite) in the OpenCMISS-Zinc package to model the maternal and fetal geometries. We then use a distributed system of Poiseuille flow equation to solve the blood flow in the arterial system. The transportation of chemical species of smoke in the arterial system is simulated by incorporating a transient advection equation. We further use an ordinary differential equation (ODE) system to simulate the time course of Carboxyhemoglobin (HbCO) in plasma during a 48 h time period. These simulation results are visualised in the arterial tree of the mother and the fetal body surface respectively for an easy understanding of the transportation process of HbCO. In conclusion a novel software tool has been developed to render scientific data in a 3D pregnant woman model and to convey educational messages for smoking cessation and other purposes.

Keywords: Pregnant woman · Fetus · Physiological simulation · Smoking cessation

1 Introduction

Smoking during pregnancy is associated with an increased risk of neonatal and infant death and low birth weight [1]. Despite of mounting evidence of such harmful effects, smoking in pregnancy is still common in some groups of pregnant women. For example, an estimated 32% of women who are Māori (the indigenous people of New Zealand) smoke during pregnancy [2]. Many developed countries

Supported by Science for Technological Innovation, New Zealand.

© Springer Nature Singapore Pte Ltd. 2020

M. Cree et al. (Eds.): ACPR 2019 Workshops, CCIS 1180, pp. 160–168, 2020.

https://doi.org/10.1007/978-981-15-3651-9_15

have strong smoke-free policies and widely available quit smoking services to help people change their smoking behaviour [3]. However, pregnant smokers are less likely to access such services and typically have lower levels of education and health literacy. Innovative approaches are required to assist this target population to quit smoking; strategies that use digital graphics and simulations can be of great interest, and thus, influence [4–6]. The motivation for this project is to integrate 3D models and physiological simulations into a software tool that facilitates pregnant woman education.

We have set two objectives for the software development. The first objective is to construct a parametric (Lagrange or Hermite) mesh for a pregnant woman model including fetus. The second objective is to make physiological simulations and incorporate scientific data for smoking cessation assistance. The simulations include blood flow in the cardiovascular system and drug transportation in it. This requires solving differential equation systems for transient drug concentrations in blood vessels. Since scientific data are usually reported in academic publications but with poor accessibility to the public, innovative approaches are required to convey the knowledge to public.

For instance, one study about the smoking effects on fetus describes the effects of carbon monoxide (CO), which is one of the most harmful components of cigarette smoke [7]. Red blood cells are responsible for carrying oxygen to different tissues by taking up oxygen from the blood flow. However, CO binds to haemoglobin with a 200-times stronger affinity than oxygen [7]. It penetrates the body through the lungs, bounds to hemoglobin to create Carboxyhemoglobin (HbCO) that diffuses throughout the whole body. Since the maternal blood is the only source of oxygen, nutrients and xenobiotic for the fetus, the HbCO reaches the fetus' bloodstream via the placenta. This leads to a decrease in oxygen delivery and may cause fetal hypoxia. Through mathematical modelling it is possible to simulate the clearance of HbCO in both the fetal and maternal blood. This paper presents our work in combining all these components, and designing an educational tool to show these physiological simulations.

2 Methods

2.1 3D Maternal and Fetal Models

We make use of a 3D pregnant woman and fetus model in the public domain (<https://www.turbosquid.com>), and retain the vertices of the polygonal quadrangles and triangles of the original models in the form of data clouds (Fig. 1a, the left panel). Then a parametric (linear Lagrange or cubic Hermite) mesh is constructed from the data cloud using a Cmgui software in the OpenCMISS package [8]. The parametric mesh allows for a representation of 3D objects with a small set of elements. For example, the final model of the fetus consists of 884 nodes and 948 elements.

The arterial tree is constructed in a similar manner, but in a 1D mesh rather than a 2D surface mesh. The arterial tree ranges from the aorta to peripheral arteries (Fig. 1b). We referred to the human anatomy [9] and the arterial radii

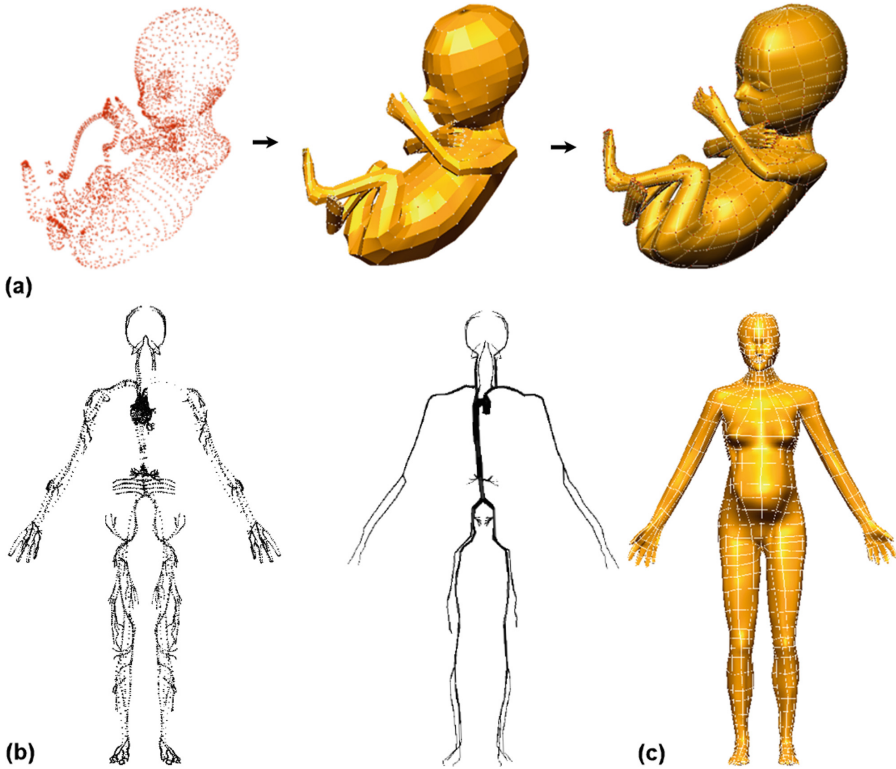


Fig. 1. Construction of the parametric mesh for the maternal and fetus model: (a) the data cloud of the fetus surface is used to create the linear Language mesh and then cubic Hermite mesh; (b) the data cloud of the cardiovascular system of the mother is used to create a parametric linear Lagrange mesh of the arterial system; (c) The cubic Hermite mesh of the maternal model.

from [10], so that the radius at each arterial node is incorporated as a computational field, then cylinders of varying diameters are swiped across the arterial tree (Fig. 1b).

Construction of the maternal surface model uses the same technique as the fetal surface model. In general, it is easier to construct a mesh for a consistent geometry such as the torso, but more difficult for a complex geometry such as the face. We referred to the cubic Hermite mesh previously made for a human body as described in [11] which was used to visualise the lymphoscintigraphy data.

2.2 Modelling the Blood Flow in the Arterial Tree

Assuming the blood flow is ruled by the Hagen-Poiseuille Law, then the pressure drop Δp is related to the flow rate q for a laminar, incompressible and Newtonian flow in a long cylindrical pipe as:

$$Q = \frac{\Delta p}{R} \quad (1)$$

where $R = 8\mu L/(\pi r^4)$ represents the vessel resistance. L and r are the length and radius for each artery. From Eq. (1) the flow velocity in a vessel is derived as $u = Q/(\pi r^2)$. For a drug in the blood flow, its concentration follows a 1D advection equation:

$$\frac{\partial C}{\partial t} + u \frac{\partial C}{\partial x} = 0 \quad (2)$$

where C is the concentration of a drug. To solve the flow across a bifurcation, we assume that the concentration is proportional to the flow rate: $C = k \times Q$ where k is a proportional constant. That assumption means that at bifurcations the concentration will follow the flow rates at daughter vessels. Equation (2) is solved using a finite difference method described in [12].

2.3 Modelling the CO Diffusion and HbCO Clearance

CO crosses the placenta to reach the fetal blood stream, causing a decrease of the amount of oxygen supplied to the fetal tissues. We use a CO exchange model between the human fetus and the mother from [7]. The simplified model allows us to simulate the HbCO clearance in the maternal and fetal plasma simultaneously. The model consists of two coupled differential equations:

$$\frac{y_1}{t} = a_{11}y_1 + f_1 \quad (3)$$

$$\frac{y_2}{t} = a_{21}y_1 + a_{22}y_2 + f_2 \quad (4)$$

where y_1 and y_2 are the concentration of HbCO in the maternal and fetal blood, respectively. a_{ij} is the diffusion constant, f_i is the function of volume distribution of CO. For details of the individual terms we refer the interested reader to [7]. By integrating the first equation, we find an expression for y_1 that can be substituted into the second equation which can be in turn integrated. Finally, the solutions are:

$$y_1 = y_1(\tau) + [y_1(0) - y_1(\tau)]e^{a_{11}t} \quad (5)$$

$$y_2 = -c_1 + c_2e^{a_{11}t} + [c_2 - y_2(0)]e^{a_{22}t} \quad (6)$$

where τ is a nominated time where the simulation ends, and is 48 h in this case.

2.4 Software Implementation

The software has been developed using the Python language. The graphic user interface (GUI) is based on the cross-platform GUI toolkit PyQt 5.4, the Python binding of Qt, which provides Qt Designer for designing and building GUIs from

Qt components. The 3D engine is based on PyZinc, the Python binding of the OpenCMISS-Zinc Application Programming Interface (API) [8]. PyZinc is an object-oriented API that consists of graphical objects defined by attributes and methods with handles to control from. The OpenCMISS package is specifically designed for physiological simulations and is the major engine behind the Physiological Human project.

Figure 1 gives an example of representing 3D bio-structures in parametric mesh. One advantage of employing this kind of mesh is that scientific data can be incorporated as *computational fields* [13]. For example, during smoking, chemicals contained in cigarette smoke move cross the blood vessels. This physical process can be simulated by colour-coding the concentration of CO, or any chemical species to provide an intuitive feel of its concentration in the tree, as we will show in the Results next.

3 Results

3.1 Physiological Simulations

The blood in the arterial system of the mother is solved from the system of differential equations (1) and (2). The simulation results are shown in Fig. 2. Here the cardiac output is assumed to be 5 L/min. The flow rate to the brain is about 800 mL/min or 16% of cardiac output. The flow to the kidney (renal perfusion) and uterus is about 1.2 L/min and 600 mL/min, or 24% and 12% of cardiac output, respectively. These are consistent with the circulation flow data in pregnant women [14]. Of specific note is that during pregnancy the renal blood flow increases drastically (up to 80%) than non-pregnant women [14].

Figure 3 visualises the concentration of [HbCO] on the fetal surface at two instants (8 h and 16 h) of a day. Equations (3)–(4) are solved under the assumption that the mother smokes about 1.5 packs of cigarette per day for 16 h. It can be seen that the [HbCO] elimination in fetus is not as quickly as the mother. [HbCO] still presents in the fetal plasma even hours after the mother stops smoking. The information could be helpful as it allows a pregnant woman to comprehend how smoking can affect the fetus.

3.2 Software Implementation

We have developed the first software for the simulation of smoking effect on 3D pregnant woman and fetus models. The GUI of the software is shown in Fig. 4. The GUI is split into two panels. The left panel contains the 3D window powered by the PyZinc engine, which renders the 3D models and physiological data. A user can navigate through the 3D environment by using the mouse. The right panel configures physiological simulations. These include the results of several models of CO exposition, based on the amount of CO the mother is exposed to during a 48 h timeframe. Once the model is loaded, the user can drag the slider under the 3D window to view the concentration of HbCO, whose colour would change from blue for the lowest concentration to red for the highest. The time and mean HbCO levels are displayed in the three corresponding boxes.

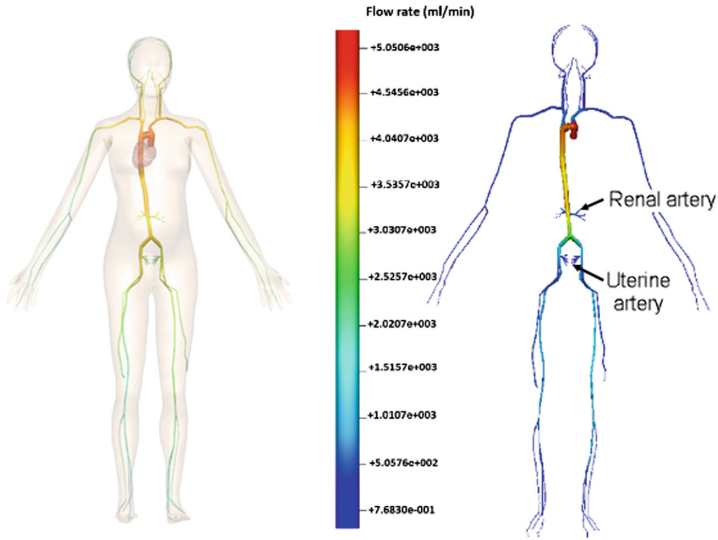


Fig. 2. Physiological simulation for the blood flow in pregnant women. The total cardiac output is 5L/min, the renal and uterine flow are 1.2L/min and 600mL/min, respectively.

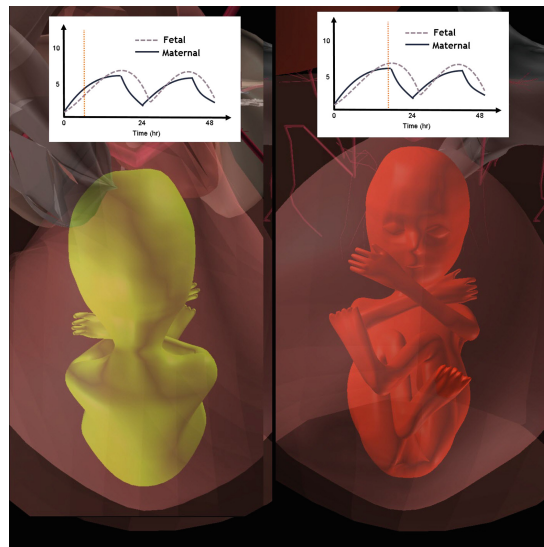


Fig. 3. Physiological simulations show that HbCO concentration in fetus (color coded on the surface). The plots are the results by solving Eqs. (4)–(5), and suggest the [HbCO] remain in the fetal blood hours after the mother stops smoking. (Color figure online)

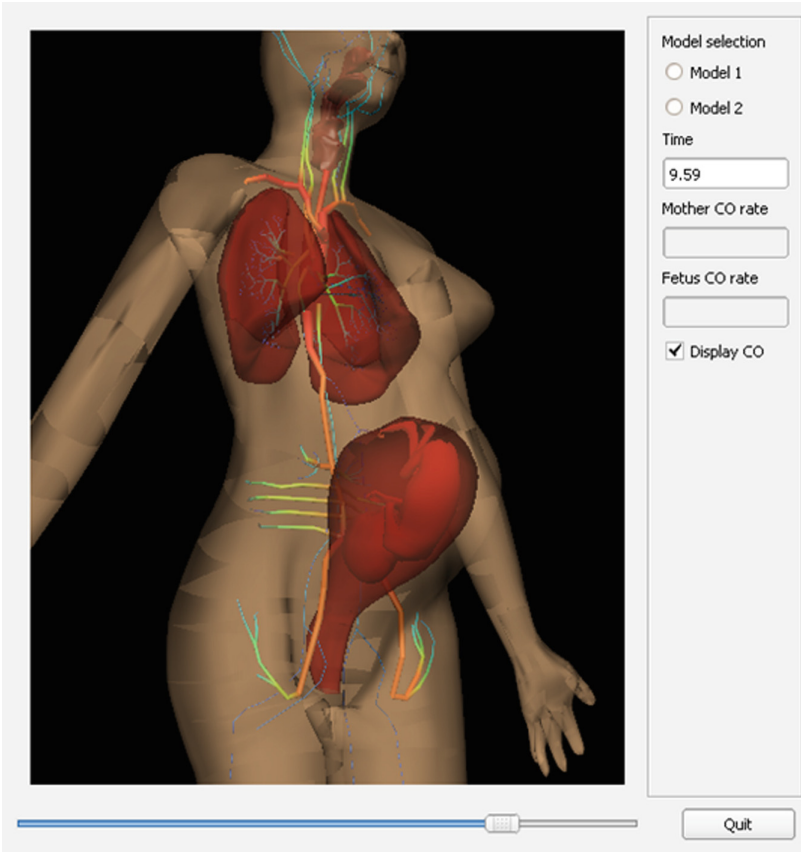


Fig. 4. The graphic user interface of the Python-based software. There are three panels in the GUI, each contains different functions as introduced in Results.

4 Discussion and Conclusion

The complications induced by cigarette smoking to fetal growth are well documented, including premature birth, delay in development, low birth weight and miscarriage [1, 15, 16]. Indeed, maternal smoking is the largest modifiable risk factor affecting fetal and infant health [5]. However, smoking cessation has proved to be difficult for many smokers. Innovative methods and novel uptake schemes need to be considered by government agencies and public health professionals. The aim of the project was to develop innovative software to address this demand and to motivate smoking pregnant women to quit smoking during pregnancy.

In the current implementation, we developed a Python-based platform that displays the 3D maternal and fetal models and visualises time-depending physiological data such as HbCO in the body. We have incorporated two physiological

simulations in the software, the first being the blood flow in the cardiovascular system of the mother, and the second being the time course simulation for HbCO. Many other physiological simulations will be made for various clinical concerns, and we leave the options to end users to advise which simulations are needed.

As the first version, there are some limitations pertaining to the software. Firstly, it would be important for incorporating more accurate anatomical structures for the simulation. An important development direction is to incorporate anatomically accurate models in particular that of the fetal model into the software. Currently the fetal model only exists in the form of a surface mesh, and no fetal organs are modelled due to a lack of fetal organ anatomy information. This is crucial for our next stage of software development, since clearance of drugs is heavily dependent on the hepatic and renal systems of fetus [17]. Secondly, we did not consider the interface between blood and air in the lungs. Neither did we consider the exchanges located in the placenta region. These could be important factors for the transport of HbCO. Thirdly when solving the transport equations for the arterial system the resistance of organs upon blood flow was not considered. A typical treatment is to connect the end of the outlets to lumped parameter models, or extend the arterial system to the level where outlet arterial pressure can be determined [18]. Lastly, the evaluation of the software system is only visual at the current stage. For the dissemination of the software, user experience feedback needs to be studied, and clinical data incorporated.

In conclusion we have developed the first 3D physiological simulation and education software for pregnant women. Feedbacks from clinicians, pregnant women and public health workers will be sought and incorporated for future developments of the software.

References

1. Kleinman, J.C., Madans, J.H.: The effects of maternal smoking, physical stature, and educational attachment of the incidence of low birth weight. *Am. J. Epidemiol.* **121**, 843–855 (1985)
2. Humphrey, G., Rossen, F., Walker, N., Bullen, C.: Parental smoking during pregnancy: findings from the growing up in New Zealand cohort. *New Zealand Med. J.* **129**, 60–74 (2016)
3. Glover, M., et al.: Driving kids to smoke? Children's reported exposure to smoke in cars and early smoking initiation. *Addict. Behav.* **36**, 1027–1031 (2011)
4. Glover, M., Fraser, T., Nosa, V.: Views of low socio-economic smokers: what will help them to quit? *J. Smok. Cessat.* **7**, 41–46 (2012)
5. McRobbie, H., Bullen, C., Glover, M., Whittaker, R., Wallace-Bell, M., Fraser, T.: New Zealand smoking cessation guidelines. *New Zealand Med. J.* **121**, 57–70 (2008)
6. Glover, M., Kira, A.: Pregnant Māori smokers' perception of cessation support and how it can be more helpful. *J. Smok. Cessat.* **7**, 65–71 (2012)
7. Hill, E.P., Hill, J.R., Power, G.G., Longo, L.D.: Carbon monoxide exchanges between the human fetus and mother: a mathematical model. *Am. J. Physiol. - Heart Circulatory Physiol.* **232**, H311–H323 (1977)

8. Bradley, C., et al.: OpenCMISS: a multi-physics & multi-scale computational infrastructure for the VPH/Physiome project. *Prog. Biophys. Mol. Biol.* **107**, 32–47 (2011)
9. Gilroy, A., MacPherson, B., Ross, L., Schuenke, M., Schulte, E., Schumacher, U.: *Atlas of Anatomy*. Thieme, New York (2008)
10. ADAN-WEB - HeMoLab (2018). <http://hemolab.lncc.br/adan-web/>
11. Reynolds, H.M., Dunbar, P.R., Uren, R.F., Blackett, S.A., Thompson, J.F., Smith, N.P.: Three-dimensional visualisation of lymphatic drainage patterns in patients with cutaneous melanoma. *Lancet Oncol.* **8**, 806–812 (2007)
12. Coutey, C., Berg, M., Ho, H., Hunter, P.: Computational simulation of blood flow and drug transportation in a large vasculature. In: Joldes, G.R.R., Doyle, B., Wittek, A., Nielsen, P.M.F.M.F., Miller, K. (eds.) *Computational Biomechanics for Medicine*, pp. 133–142. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-28329-6_12
13. Bradley, C., Pullan, A., Hunter, P.: Geometric modeling of the human torso using cubic hermite elements. *Ann. Biomed. Eng.* **25**, 96–111 (1997)
14. Sharma, R.P., Schuhmacher, M., Kumar, V.: The development of a pregnancy PBPK Model for Bisphenol A and its evaluation with the available biomonitoring data. *Sci. Total Environ.* **624**, 55–68 (2018)
15. Morgan, D.J.: Drug disposition in mother and foetus. *Clin. Exp. Pharmacol. Physiol.* **24**, 869–873 (1997)
16. Walsh, R.A.: Effects of maternal smoking on adverse pregnancy outcomes: examination of the criteria of causation. *Hum. Biol.* **66**, 1059–1092 (1994)
17. Ring, J.A., Ghabrial, H., Ching, M.S., Smallwood, R.A., Morgan, D.J.: Fetal hepatic drug elimination. *Pharmacol. Ther.* **84**, 429–445 (1999)
18. Muller, A., Clarke, R., Ho, H.: Fast blood-flow simulation for large arterial trees containing thousands of vessels. *Comput. Methods Biomech. Biomed. Eng.* **20**, 160–170 (2017)



Resolution Conversion of Volumetric Array Data for Multimodal Medical Image Analysis

Kento Hosoya¹, Kouki Nozawa¹, and Atsushi Imiya²(✉)

¹ School of Science and Engineering, Chiba University,
Yayoi-cho 1-33, Inage-ku, Chiba 263-8522, Japan

² Institute of Management and Information Technologies, Chiba University,
Yayoi-cho 1-33, Inage-ku, Chiba 263-8522, Japan
imiya@faculty.chiba-u.jp

Abstract. This paper aims to clarify statistical and geometric properties of linear resolution conversion for registration between different resolutions observed using the same modality. The pyramid transform for higher-dimensional array with rational-order is formulated by means of tensor decomposition. For fast processing of volumetric data, compression of data is an essential task. Three-dimensional extension of the pyramid transform reduces the sizes of the volumetric data by factor 2. Extension of matrix expression of the pyramid transform to the operation of tensors using the mode product of a tensor and matrix derives the pyramid transform for volumetric data of the rational orders. The pyramid transform is achieved by downsampling after linear smoothing. The dual operation of the pyramid transform is achieved by linear interpolation after upsampling. The rational-order pyramid transform is decomposed into upsampling by linear interpolation and the traditional pyramid transform with the integer order. By controlling ratio between upsampling for linear interpolation and downsampling in the pyramid transform, the rational-order pyramid transform is computed. The tensor expression of the volumetric pyramid transform clarifies that the transform yields the orthogonal base systems for any ratios of the rational pyramid transform.

1 Introduction

There are modern demands on tensor-based higher-dimensional visual processing from medical image analysis [12], microstructure analysis of material science and biology [1] and visualisation of natural phenomena [16]. Objects and phenomenon in natures including human cells and organs are fundamentally observed and described as a spatio-temporal volumetric sequences. Therefore, even snapshots of them in a temporal series are three-dimensional data. For fast processing of volumetric data, compression of data is an essential task [16, 17]. Registration between images with the same resolution observed by the same modality is a standard framework [22, 23]. The second class of problems is registration between

images observed using different modalities [9, 24, 25]. The third one is registration between different resolutions observed using the same modality [6]. This paper focuses on linear resolution conversion for the third problem.

In this paper, the rational-order pyramid transform [2–7] for volumetric array data is formulated by means of tensor decomposition. We 30 mm the matrix expression of the pyramid transform for discrete signals. In Ref. [2], the rational order pyramid transform is designed from the viewpoint of synthesis of IIR filter banks by assuming a biorthogonal relation for the kernels of pyramid transform and its dual transform. The matrix-based expression, however, yields orthogonal bases in each resolution. Employing the tensor expression of higher-dimensional array data and mode product of a tensor and matrix, we introduce the pyramid transform for volumetric array data. The pyramid transform is a classical method for multiresolution image analysis. Multiresolution image analysis establishes stable and accurate feature extraction by transmitting global features in a coarse resolution to local and precise features in a fine resolution. The rational order pyramid transform reduces the size of the volumetric data for any ratio. Furthermore, tensor expression of the volumetric pyramid transform clarifies that the transform yields orthogonal base systems for any sizes of scale reduction by the rational pyramid transform.

For longitudinal analysis [8], the registration of a temporal sequence of images observed by different modalities [9–12] is demanded. The resolution of medical images depends on the modalities of the observations. Even if the same physical observation modality is used for measuring each image in a sequence, images with different resolutions are measured. For instance, the width of the x-ray used in computerised tomography affects resolution of slice images. The same slice images measured by the x-rays with different energies possess the different resolutions [13], since the energy of the x-ray mathematically defines the width of the x-ray beams. The registration of reference and target images [14, 15, 23] with different resolutions is an essential problem in medical image registration. For the normalisation of resolutions for the registration of an image pair, the pyramid transform [3–7] of rational orders is demanded.

2 Mathematical Preliminaries

2.1 3D Signal Processing

We assume that our images are elements of the Sobolev space $H^2(\mathbf{R}^3)$. We define the linear transforms

$$g(\mathbf{u}) = Rf(\mathbf{u}) = \int_{\mathbf{R}^3} w_\sigma(\mathbf{x})f(\sigma\mathbf{u} - \mathbf{x})d\mathbf{x}, \quad (1)$$

$$f(\mathbf{x}) = Eg(\mathbf{x}) = \frac{1}{\sigma^3} \int_{\mathbf{R}^3} w_\sigma(\mathbf{u})g\left(\frac{\mathbf{x} - \mathbf{u}}{\sigma}\right) d\mathbf{u}, \quad (2)$$

where $w_\sigma(\mathbf{x}) = w_\sigma(x)w_\sigma(y)w_\sigma(z)$, for $\mathbf{x} = (x, y, z)^\top \in \mathbf{R}^3$.

Definition 1. In both the domain and range spaces of the transform R , the inner products of functions are defined as

$$(f, g)_D = \int_{\mathbf{R}^3} f(\mathbf{x})g(\mathbf{x})d\mathbf{x}, \quad (Rf, Rg)_R = \int_{\mathbf{R}^3} Rf(\mathbf{u})Rg(\mathbf{u})d\mathbf{u}. \quad (3)$$

The dual operation R^* of the operation R satisfies the relation $(f, Rg)_R = (R^*f, g)_D$.

Since for the operations R and E , the relation

$$\int_{\mathbf{R}^3} Rf(\mathbf{u})g(\mathbf{u})d\mathbf{u} = \int_{\mathbf{R}^3} f(\mathbf{x})Eg(\mathbf{x})d\mathbf{x} \quad (4)$$

is satisfied, we have the relation $R^* = E$.

For $\sigma > 0$, dilation filtering is

$$h(\mathbf{x}) = g(\mathbf{x}) *_{\sigma} f(\mathbf{x}) = \int_{\mathbf{R}^3} g(\mathbf{x} - \sigma\mathbf{y})f(\mathbf{y})d\mathbf{y} = \frac{1}{\sigma^3} \int_{\mathbf{R}^3} g(\mathbf{u})f\left(\frac{\mathbf{x} - \mathbf{u}}{\sigma}\right) d\mathbf{u}. \quad (5)$$

This equation coincides with Eq. (2) if we set $w_{\sigma}(\mathbf{x}) = g(\mathbf{x})$. The discrete dilation filtering of factor k for sequence [18] is

$$h_i = f_i *_{k} g_i = \sum_{m+kn=i} g_m f_n = \sum_{j=-\infty}^{\infty} g_{i-kj} f_j = \sum_{j=-\infty}^{\infty} g_j f_{\frac{j-i}{k}}. \quad (6)$$

assuming that summation is achieved for $j - i = kp$.

For the three-dimensional array $f_{ijk} = f(\Delta i, \Delta j, \Delta k)$, which is the samples of $f(\mathbf{x})$ on $\mathbf{x} = (\Delta i, \Delta j, \Delta k)^T$ for $\mathbf{z} \in \mathbf{Z}^3$, the volumetric pyramid transform of the order p is

$$g_{mnk} = h_{pm pn pk},$$

$$h_{mnk} = \sum_{\alpha, \beta, \gamma = -(p-1)}^{(p-1)} \frac{p - |\alpha|}{p^2} \cdot \frac{p - |\beta|}{p^2} \cdot \frac{p - |\gamma|}{p^2} f_{m+\alpha n+\beta k+\gamma}. \quad (7)$$

The dual transform is

$$f_{pm+\alpha pn+\beta pk+\gamma} = \frac{1}{p^3} \left(\frac{p - \alpha}{p} \cdot \frac{p - \beta}{p} \cdot \frac{p - \gamma}{p} g_{pm pn pk} + \frac{\alpha}{p} \cdot \frac{\beta}{p} \cdot \frac{\gamma}{p} g_{p(m+1) p(n+1) p(k+1)} \right), \quad (8)$$

for $\alpha, \beta, \gamma = 0, 1, \dots, (p - 1)$.

2.2 Tensor Algebra and Decomposition

For the triplet of positive integers I_1, I_2 and I_3 , the third-order tensor $\mathbf{R}^{I_1 \times I_2 \times I_3}$ is expressed as $\mathcal{X} = ((x_{ijk}))$. Indices i, j and k are called the 1-mode, 2-mode

and 3-mode of \mathcal{X} , respectively. The tensor space $\mathbf{R}^{I_1 \times I_2 \times I_3}$ is interpreted as the Kronecker product of three vector spaces \mathbf{R}^{I_1} , \mathbf{R}^{I_2} and \mathbf{R}^{I_3} such that $\mathbf{R}^{I_1} \otimes \mathbf{R}^{I_2} \otimes \mathbf{R}^{I_3}$. We set $I = \max(I_1, I_2, I_3)$.

Samples $Sf(\Delta \mathbf{z})$ for $|\mathbf{z}|_\infty \leq I$ yield an $I \times I \times I$ three-way array \mathbf{F} . To preserve the multi-linearity of the function $f(\mathbf{x})$, we deal with the array \mathbf{F} as a third-order tensor \mathcal{F} . The operation $\text{vec}\mathcal{F}$ derives a vector $\mathbf{f} \in \mathbf{R}^{I_{123}}$ for $I_{123} = I_2 \cdot I_2 \cdot I_3$. We can reconstruct f from \mathcal{F} using an interpolation procedure.

For \mathcal{X} , the n -mode vectors, $n = 1, 2, 3$, are defined as the I_n -dimensional vectors obtained from \mathcal{X} by varying this index i_n while fixing all the other indices.

The unfolding of \mathcal{X} along the n -mode vectors of \mathcal{X} is defined as matrices such that $\mathcal{X}_{(1)} \in \mathbf{R}^{I_1 \times I_{23}}$, $\mathcal{X}_{(2)} \in \mathbf{R}^{I_2 \times I_{13}}$ and $\mathcal{X}_{(3)} \in \mathbf{R}^{I_3 \times I_{12}}$ for $I_{12} = I_1 \cdot I_2$, $I_{23} = I_2 \cdot I_3$ and $I_{13} = I_1 \cdot I_3$, where the column vectors of $\mathcal{X}_{(j)}$ are the j -mode vectors of \mathcal{X} for $i = 1, 2, 3$. We express the j -mode unfolding of \mathcal{X}_i as $\mathcal{X}_{i,(j)}$.

For matrices

$$\mathbf{U} = ((u_{ii'})) \in \mathbf{R}^{I_1 \times I_1}, \quad \mathbf{V} = ((v_{jj'})) \in \mathbf{R}^{I_2 \times I_2}, \quad \mathbf{W} = ((w_{kk'})) \in \mathbf{R}^{I_3 \times I_3}, \quad (9)$$

the n -mode products for $n = 1, 2, 3$ of a tensor \mathcal{X} are the tensors with entries

$$\begin{aligned} (\mathcal{X} \times_1 \mathbf{U})_{ijk} &= \sum_{i'=1}^{I_1} x_{i'jk} u_{i'i}, \\ (\mathcal{X} \times_2 \mathbf{V})_{ijk} &= \sum_{j'=1}^{I_2} x_{ij'k} v_{j'j}, \\ (\mathcal{X} \times_3 \mathbf{W})_{ijk} &= \sum_{k'=1}^{I_3} x_{ijk'} w_{k'k}, \end{aligned} \quad (10)$$

where $(\mathcal{X})_{ijk} = x_{ijk}$ is the ijk -th element of the tensor \mathcal{X} . The inner product of two tensors \mathcal{X} and \mathcal{Y} in $\mathbf{R}^{I_1 \times I_2 \times I_3}$ is

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i=1}^{I_1} \sum_{j=1}^{I_2} \sum_{k=1}^{I_3} x_{ijk} y_{ijk}. \quad (11)$$

Using this inner product, we have the Frobenius norm of a tensor \mathcal{X} as $|\mathcal{X}|_F = \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle}$. The Frobenius norm $|\mathcal{X}|_F$ of the tensor \mathcal{X} satisfies the relation $|\mathcal{X}|_F = |\mathbf{f}|_2$, where $|\mathbf{f}|_2$ is the Euclidean norm of the vector \mathbf{f} . If $\mathcal{X} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c} = ((x_{ijk}))$, $x_{ijk} = a_i b_j c_k$ for $\mathbf{a} = (a_1, a_2, \dots, a_{n_1})^\top$, $\mathbf{b} = (b_1, b_2, \dots, b_{n_2})^\top$ and $\mathbf{c} = (c_1, c_2, \dots, c_{n_3})^\top$.

For the three-dimensional array $\mathcal{F} = ((f_{ijk})) \in \mathbf{R}^{n_1 \times n_2 \times n_3}$, the mode products of tensors with matrices $\mathbf{A} = ((a_{ij})) \in \mathbf{R}^{m_1 \times n_1}$, $\mathbf{B} = ((b_{ij})) \in \mathbf{R}^{m_2 \times n_2}$ and $\mathbf{C} = ((c_{ij})) \in \mathbf{R}^{m_3 \times n_3}$ are defined as

$$\mathcal{F} \times_1 \mathbf{A}^\top = \sum_{\alpha=1}^{n_1} f_{\alpha j k} a_{i\alpha}, \quad \mathcal{F} \times_2 \mathbf{B}^\top = \sum_{\beta=1}^{n_2} f_{i \beta k} b_{j\beta}, \quad \mathcal{F} \times_3 \mathbf{C}^\top = \sum_{\gamma=1}^{n_3} f_{i j \gamma} c_{k\gamma}. \quad (12)$$

The shift-invariant operation is expressed as

$$g(x, y, z) = \int \int \int_{\Omega} a(u-x)b(v-y)c(w-z)f(u, v, w)dudvdw. \quad (13)$$

We assume that all $a(x)$, $b(y)$, $c(z)$ and $f(x, y, z)$ are zero outside of a finite support Ω . The discrete shift-invariant operation

$$g_{ijk} = \sum_{\alpha=-\infty}^{\infty} \sum_{\beta=-\infty}^{\infty} \sum_{\gamma=-\infty}^{\infty} a_{\alpha-i}b_{\beta-j}c_{\gamma-k}f_{\alpha\beta,\gamma} \quad (14)$$

with a finite support is expressed as

$$\mathbf{G} = \mathbf{F} \times_1 \mathbf{A}^\top \times_2 \mathbf{B}^\top \times_3 \mathbf{C}^\top, \quad (15)$$

where $a_{ij} = a_{|i-j|}$, $b_{ij} = b_{|i-j|}$ and $c_{ij} = c_{|i-j|}$, using the mode product of tensor. Using the discrete cosine transform (DCT) matrix of the type II

$$\Phi = \left(\left(\frac{1}{\sqrt{n}} s_j \cos \frac{(2j+1)i}{2n} \pi \right) \right) = (\varphi_0, \varphi_1, \dots, \varphi_{n-1}), \quad s_j = \begin{cases} 1, & \text{if } j = 0, \\ \frac{1}{\sqrt{2}}, & \text{otherwise,} \end{cases} \quad (16)$$

the DCT for three-dimensional array is expressed as

$$\mathcal{G} = \mathcal{F} \times_1 \Phi^\top \times_2 \Phi^\top \times_3 \Phi^\top. \quad (17)$$

3 Rational-Order Pyramid Transform

3.1 Eigenspace Analysis of Pyramid Transform of Sequences

The pyramid transform

$$g_n := \frac{1}{4}f_{2n-1} + \frac{1}{2}f_{2n} + \frac{1}{4}f_{2n+1} = \frac{1}{4}(f_{2n-1} + 2f_{2n} + f_{2n+1}) \quad (18)$$

for the sequence $\{f_n\}_{n=-\infty}^{\infty}$ is redescribed as

$$g_n = h_{2n}, \quad h_n = \frac{1}{4}(f_{n-1} + 2f_n + f_{n+1}) = f_n + \frac{1}{2} \left(\frac{f_{n-1} - 2f_n + f_{n+1}}{2} \right). \quad (19)$$

These relations imply that the pyramid transform is achieved by downsampling after computing moving average.

For the Neumann boundary condition, the one-dimensional discrete Laplacian \mathbf{L} is

$$\mathbf{L} = \frac{1}{2}\mathbf{D}, \quad \mathbf{D} = \begin{pmatrix} -1 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 1 & -2 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 & -1 \end{pmatrix}. \quad (20)$$

The eigendecomposition of \mathbf{D} is

$$\mathbf{D}\boldsymbol{\Phi} = \boldsymbol{\Phi}\boldsymbol{\Lambda}, \quad \boldsymbol{\Lambda} = ((\lambda_k \delta_{kl})) \quad \lambda_k^{(n)} = 4 \sin^2 \frac{\pi k}{2n}. \quad (21)$$

The matrix of downsampling operation for vectors is

$$\mathbf{S}_q = \mathbf{I} \otimes \mathbf{e}_1^q, \quad \mathbf{e}_1^q = (1, 0, \dots, 0)^\top \in \mathbf{R}^q. \quad (22)$$

Furthermore, the $2p + 1$ -dimensional diagonal matrix

$$\mathbf{N}_p = ((n_{|i-j|})), \quad n_k = \frac{p-k}{p}, \quad 0 \leq p \leq k \quad (23)$$

is expressed as

$$\mathbf{N}_p = \sum_{k=0}^p a_k \mathbf{D}^k, \quad \mathbf{D}^0 = \mathbf{I}, \quad (24)$$

for an appropriate collection of coefficients $\{a_k\}_{k=1}^p$. Using matrices \mathbf{N}_p and \mathbf{S}_p , the linear interpolation for order p is expressed

$$\mathbf{L}_p = \mathbf{N}_p \mathbf{S}_p. \quad (25)$$

as the matrix. Equation (25) implies the following property.

Property 1. Assuming that the domain of signals is $\mathcal{L}\{\varphi_i\}_{i=0}^{n-1}$, the range of signals upsampled using linear interpolation of order p is $\mathcal{L}\{\varphi_i\}_{i=0}^{pn-1}$.

Using the relation in Eq. (19), the pyramid transform of order q is expressed as

$$\mathbf{R}_q = \frac{1}{q} \mathbf{S}_q \mathbf{N}_q, \quad (26)$$

since the pyramid transform is achieved by downsampling after shift-invariant smoothing, for which the matrix expression is \mathbf{N}_q . Equation (26) implies the following theorem.

Theorem 1. *With the Neumann boundary condition, the pyramid transform of order q is a linear transform from $\mathcal{L}\{\varphi_i\}_{i=0}^{n-1}$ to $\mathcal{L}\{\varphi_i\}_{i=0}^{\frac{1}{q}n-1}$, assuming that $n = kq$.*

Equations (25) and (26) derive the following theorem.

Theorem 2. *The q/p -pyramid transform is expressed as*

$$\mathbf{R}_{q/p} = \frac{1}{q} \mathbf{S}_q \mathbf{N}_q \mathbf{L}_p = \frac{1}{q} \mathbf{S}_q \mathbf{N}_{q/p} \mathbf{S}_p, \quad \mathbf{N}_{q/p} = \mathbf{N}_q \mathbf{N}_p. \quad (27)$$

Theorem 2 implies the following theorem.

Theorem 3. *With the Neumann boundary condition, the q/p -pyramid transform is a linear transform from $\mathcal{L}\{\varphi_i\}_{i=0}^{n-1}$ to $\mathcal{L}\{\varphi_i\}_{i=0}^{\frac{q}{p}n-1}$.*

Since the matrix expression of $R_{q/p}^*$ and $E_{q/p}$ are $\mathbf{R}_{q/p}^\top$ and $\mathbf{E}_{q/p}^\top$, respectively, we have the following theorem.

Theorem 4. *For a rational number q/p , the pyramid transform and its dual transform satisfy the relations,*

$$\mathbf{R}_{q/p}^* = \mathbf{R}_{q/p}^\top = \mathbf{E}_{q/p} = \mathbf{R}_{p/q}, \quad \mathbf{E}_{q/p}^* = \mathbf{E}_{q/p}^\top = \mathbf{R}_{q/p} = \mathbf{E}_{p/q}, \quad (28)$$

where \mathbf{A}^* is the dual operation of the linear transform \mathbf{A} .

Theorem 4 implies the following theorem.

Theorem 5. *If $\mathbf{R}_{q/p}^\top = \mathbf{E}_{s/r}$, the relation $q/p \times s/r = 1$ is satisfied for the rational-number pair q/p and s/r .*

The linear scale space transform is used for multiresolution image analysis. For a pair of positive number such that $p + q = 1$, the relation

$$\binom{n}{k} \sim \frac{1}{\sqrt{2\pi npq}} \exp\left(-\frac{(k - np)^2}{2npq}\right) \quad (29)$$

is called De Moivre-Laplace theorem. This theorem implies the discrete approximation of the linear scale space transform using the binomial distribution.

Table 1 summarises the relations between p -pyramid transform for positive integers and the scaled linear scale-space transform for signals. This table clarifies the relation between the signals yielded by the p -pyramid transform and numerically approximated linear scale space transform.

Table 1. Relations between the pyramid transform and linear scale-space transform.

Discrete expression	Continuous expression
$g_m = \sum_{k=-1}^1 w_k f_{2m-k}$	$g(x) = \int_{-\infty}^{\infty} w_2(y) f(2y - x) dx$
$g_m = \sum_{k=-n}^n \frac{1}{(2n)!} \binom{2n}{n-k} f_{m-k}$	$g(x) = \frac{1}{\sqrt{2\pi\tau}} \int_{-\infty}^{\infty} \exp\left(-\frac{y^2}{2\tau}\right) f(x - y) dy$
$g_m = h_{qm}$	$g(x) = \int_{-\infty}^{\infty} w_\sigma(y) f(\sigma y - x) dx, \sigma > 0$
$h_n = \sum_{\alpha=-(q-1)}^{(q-1)} \frac{p - \alpha }{q^2} k_{n+\alpha}$	$w_\sigma(x) = \begin{cases} \frac{1}{\sigma} (1 - \frac{1}{\sigma} x), & x \leq \sigma \\ 0, & x > \sigma \end{cases}$
$k_{m+\beta} = \frac{1}{p} \left(\frac{p-\beta}{p} f_m + \frac{\beta}{p} f_{m+1} \right)$	
$\mathbf{f}^{(k)} = \left(\Phi \left(\mathbf{I} - \frac{\tau}{2} \mathbf{A} \right)^{-k} \Phi^\top \right) \mathbf{f}$ (see Appendix.)	$\frac{\partial}{\partial \tau} f(x, \tau) = \frac{1}{2} \frac{\partial^2}{\partial x^2} f(x, \tau), \tau \geq 0$
$\mathbf{f}^{(k)} = (f_0^{(k)}, f_1^{(k)}, \dots, f_{n-1}^{(k)})^\top$	

3.2 Eigenspace Analysis of Pyramid Transform of 3D Array

Using the relation between Eqs. (7) and (8), we construct the pyramid transform of the rational order q/p for $p, q \in \mathbf{Z}_+$.

Definition 1. *The q/p pyramid transform first achieves upsampling of order p by using linear interpolation. For the upsampled data, the pyramid transform of order q is applied.*

We call the transform the q/p -pyramid transform.

Definition 2. *The dual transform is achieved by downsampling to the result of the dual transform of the pyramid transform.*

Equation (8) is the linear interpolation of $g_{pm\,pn\,pk}$ to generate $f_{m\,n\,k}$ for $k, m, n = 0, \pm 1 \dots, \pm \infty$.

The pyramid transform for three-dimensional volumetric digital images is

$$\mathcal{G} = \mathcal{F} \times_1 \mathbf{R}_{q/p}^\top \times_2 \mathbf{R}_{q/p}^\top \times_3 \mathbf{R}_{q/p}^\top. \tag{30}$$

The transform allows us to compress the volumetric data to a small size.

Equation (30) implies the following properties and theorem, where $\varphi_{ijk}^{\alpha\beta\gamma} = \varphi_{\alpha i} \varphi_{\beta j} \varphi_{\gamma k}$ for

$$\mathcal{U} = \varphi_\alpha \circ \varphi_\beta \circ \varphi_\gamma = ((\varphi_{ijk}^{\alpha\beta\gamma})). \tag{31}$$

Property 2. Assuming that the domain of images is $\mathcal{L}\{\varphi_i \circ \varphi_j \circ \varphi_k\}_{i,j,k=0}^{n-1}$ for $n = 2^m$, the range of subsampled images by order p is $\mathcal{L}\{\varphi_i \circ \varphi_j \circ \varphi_k\}_{i,j,k=0}^{pn-1}$.

Property 3. With the Neumann boundary condition, the pyramid transform of order q is a linear transform from $\mathcal{L}\{\varphi_i \circ \varphi_j \circ \varphi_k\}_{i,j,k=0}^{n-1}$, to $\mathcal{L}\{\varphi_i \circ \varphi_j \circ \varphi_k\}_{i,j,k=0}^{\frac{1}{q}n-1}$, assuming $n = lq$.

Theorem 6. *With the Neumann boundary condition, the q/p pyramid transform is a linear transform from $\mathcal{L}\{\varphi_i \circ \varphi_j \circ \varphi_k\}_{i,j,k=0}^{n-1}$, to $\mathcal{L}\{\varphi_i \circ \varphi_j \circ \varphi_k\}_{i,j,k=0}^{\frac{p}{q}n-1}$.*

Since the vectors $\{\varphi_i\}_{i=0}^{n-1}$ satisfies the relation $\varphi_i^\top \varphi_j = \delta_{ij}$, the relation

$$\langle \varphi_i \circ \varphi_j \circ \varphi_k, \varphi_{i'} \circ \varphi_{j'} \circ \varphi_{k'} \rangle = \delta_{i i'} \delta_{j j'} \delta_{k k'}. \tag{32}$$

is satisfied. Therefore, the q/p -pyramid transform derives an orthogonal base system for q/p .

4 Statistical Property

For a positive function $f(x, y, z) \geq 0$ defined on the domain $\Omega = [0, a] \times [0, b] \times [0, c]$, we define the function

$$f(x, y, z; u) = \begin{cases} f(x, y, z), & \text{if } f(x, y, z) < u, \\ u, & \text{otherwise.} \end{cases} \quad (33)$$

Since the total value of $f(x, y, z)$ smaller than u is

$$U(u; f(x, y, z)) = \int_0^a \int_0^b \int_0^c f(x, y, z; u) dx dy dz, \quad (34)$$

the total value for $f(x, y, z) = u$ is computed as

$$\begin{aligned} H(u; f(x, y, z)) &= \lim_{\delta \rightarrow 0} \frac{U(u + \delta; f(x, y, z)) - U(u - \delta; f(x, y, z))}{2\delta} \\ &= \frac{\partial}{\partial u} U(u; f(x, y, z)). \end{aligned} \quad (35)$$

The function

$$h(u; f(x, y, z)) = \frac{1}{|a \times b \times c|} \frac{\partial}{\partial u} U(u; f(x, y, z)) \quad (36)$$

is the normalised grey-value histogram of $f(x, y, z)$ on Ω .

The distance between a pair of normalised histograms $p(u) = h(u; f(x, y, z))$ and $q(u) = h(u; g(x, y, z))$ is computed by the transportation

$$d_\alpha(p, q) = \min_{c(x, y)} \sqrt[\alpha]{\int_0^{u_{\max}} \int_0^{v_{\max}} |p(u) - q(v)|^\alpha c(u, v) du dv} \quad (37)$$

for $\alpha > 0$ with the conditions

$$\int_0^{u_{\max}} c(u, v) du \leq q(v), \quad \int_0^{v_{\max}} c(u, v) dv \leq p(u).$$

Resolutions of original images in Figs. 1 and 2 are $362 \times 434 \times 362$ and $111 \times 102 \times 159$ voxels, respectively. We assume that volumetric images are defined in the region

$$\Omega = [-w, w] \times [-h, h] \times [-d, d]$$

and that the centroids of images are aligned to the origin for each resolution. The grey-value histograms are generated from voxel values in the region

$$\Omega^{1/2} = \left[-\frac{w}{2}, \frac{w}{2}\right] \times \left[-\frac{h}{2}, \frac{h}{2}\right] \times \left[-\frac{d}{2}, \frac{d}{2}\right]$$

in each resolution. The result of q/p -pyramid transform is defined in

$$\Omega_{q/p} = \left[-\frac{q}{p} \cdot w, \frac{q}{p} \cdot w \right] \times \left[-\frac{q}{p} \cdot h, \frac{q}{p} \cdot h \right] \times \left[-\frac{q}{p} \cdot d, \frac{q}{p} \cdot d \right].$$

The grey-value histogram is generated from voxel values in

$$\Omega_{q/p}^{1/2} = \left[-\frac{q}{p} \cdot \frac{w}{2}, \frac{q}{p} \cdot \frac{w}{2} \right] \times \left[-\frac{q}{p} \cdot \frac{h}{2}, \frac{q}{p} \cdot \frac{h}{2} \right] \times \left[-\frac{q}{p} \cdot \frac{d}{2}, \frac{q}{p} \cdot \frac{d}{2} \right].$$

These operations reduce the number of voxels in the background for the generation of grey-value histogram, since the grey-values on the background voxels causes biases on grey-value distribution in histograms. The top and bottom matrices in Table 2 show the distances among the images in Figs. 1 and 2, respectively, for $\alpha = 2$. In these matrices, the elements in the upper triangles are computed, since the matrices are symmetric.

The numerical experiments imply that the grey-value distributions of the results of the volumetric q/p -pyramid transform possess the same distribution property as the original images, that is, the transform preserves the shapes of grey-value histograms of images.

Setting $D(q/p, r/s)$ to be the transportation distance between a pair of the normalised grey-value histograms of images computed by the q/p - and r/s -pyramid transforms, Table 2 shows that

$$|D(q/p, r/s) - D(b/a, f/e)| \leq C \tag{38}$$

for a positive number C and any combinations of four rational numbers q/p , r/s , b/a and f/e .

Table 2. Distance matrix of the normalised grey-value histograms for the volumetric brain images for $\alpha = 2$.

	1/1	1/2	1/3	2/3
1/1	0	2.88×10^{-3}	7.21×10^{-3}	2.89×10^{-3}
1/2	*	0	1.24×10^{-3}	1.19×10^{-5}
1/3	*	*	0	1.21×10^{-3}
2/3	*	*	*	0

	1/1	1/2	1/3	2/3
1/1	0	1.15×10^{-3}	3.16×10^{-3}	2.05×10^{-3}
1/2	*	0	1.17×10^{-3}	0.34×10^{-5}
1/3	*	*	0	0.71×10^{-3}
2/3	*	*	*	0

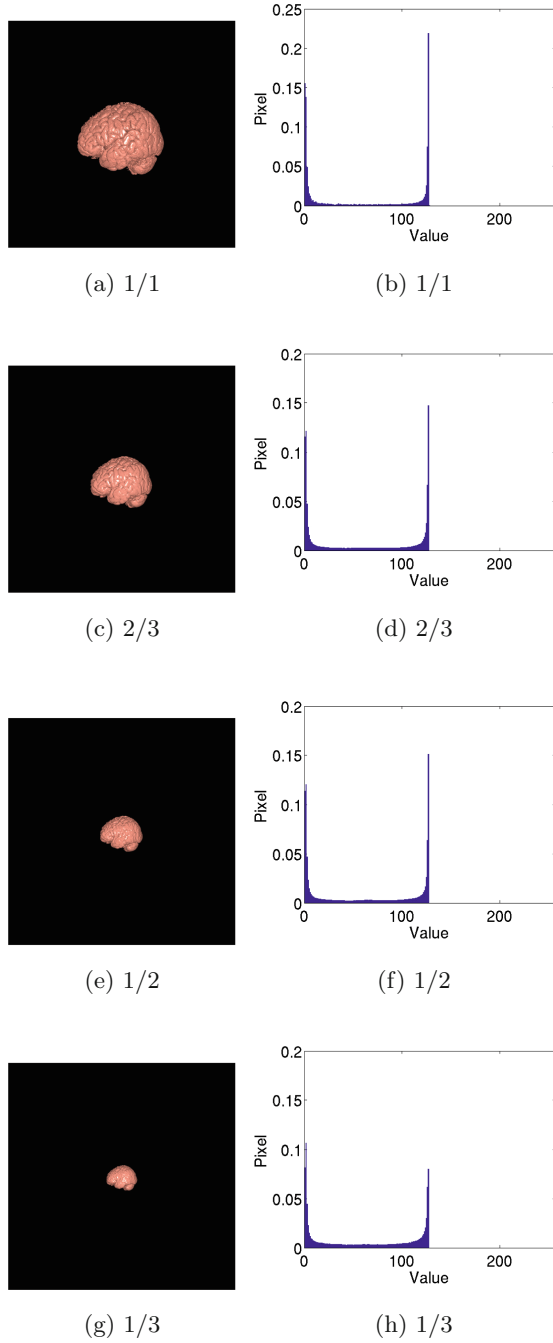


Fig. 1. Pyramid transform of images and their grey-value histograms. Resolution of the original volumetric brain images [19] is $362 \times 434 \times 362$.

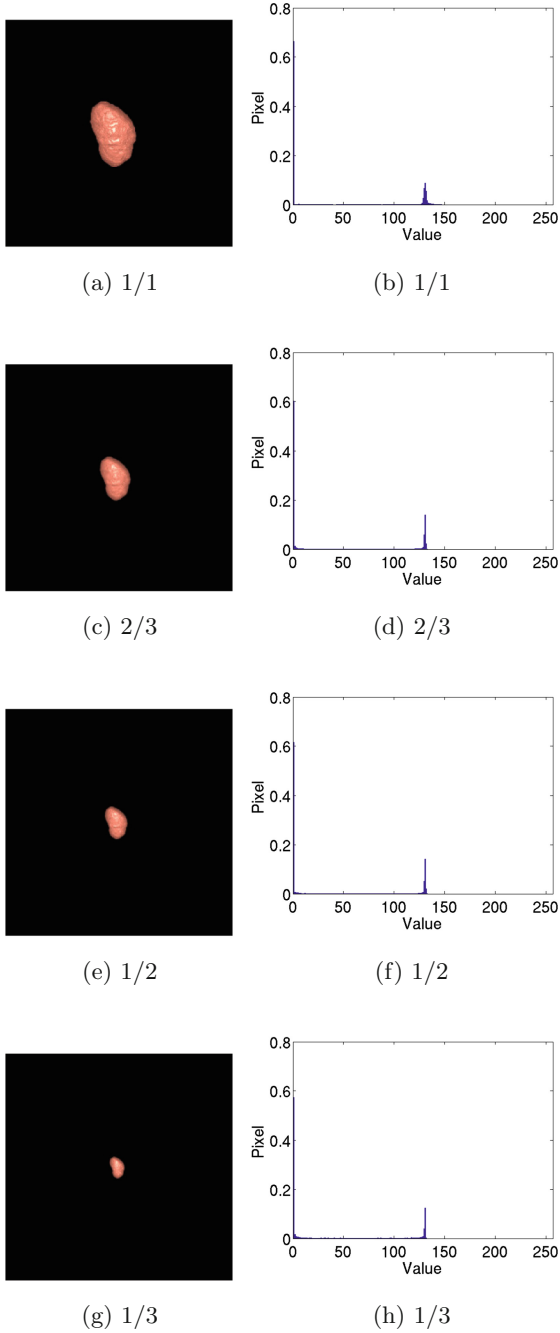


Fig. 2. Pyramid transform of images and their grey-value histograms. Resolution of the original volumetric kidney image is $111 \times 102 \times 159$

5 Conclusions

We have introduced a tensor-based framework for the rational-order pyramid transform of volumetric data, which we call the q/p pyramid transform. The rational-order pyramid transform reduces the size of images by any rational factors. The transform corresponds to the traditional transform if $q/p = 1/2$. If the result of the pyramid transform is expressed in the same landscape with the original images, the result of the transform yields a low-resolution image in any rational order. Numerical experiments imply that the grey-value distributions of the results of the q/p pyramid transform processes the same distribution property with the original image, that is, the transform preserves the shapes of grey-scale histograms of images. Furthermore, tensor expression of the volumetric pyramid transform clarifies that the transform yields the orthogonal base systems for any ratios of the rational pyramid transform.

Since the dual transform of the pyramid transform is achieved by convolution after upsampling, the dual transform to the pyramid transform is dilated convolution with the triangle kernel [18]. Since the statistical properties of grey-value histogram are fulfilled to the dual transform of the pyramid transform, the dilated convolution with the triangle kernel preserves the statistical properties of grey-values through factors of dilation.

Registration between images with the same resolution observed by the same modality is a standard framework [22, 23]. The second class of problems is registration between images observed using different modalities [9, 24, 25]. The third one is registration between different resolutions observed using the same modality [6]. This paper focused on the third problem. Numerical and statistical experiments showed that the rational order pyramid transform is used as a resolution-conversion method for multi-modernity image registration.

Acknowledgements. This research was supported by the “Multidisciplinary Computational Anatomy and Its Application to Highly Intelligent Diagnosis and Therapy” project funded by a Grant-in-Aid for Scientific Research on Innovative Areas from MEXT, Japan, and by Grants-in-Aid for Scientific Research funded by the Japan Society for the Promotion of Science.

Appendix: Discrete Heat Equation

For the heat equation $\frac{\partial f}{\partial \tau} = \frac{1}{2} \cdot \frac{\partial^2 f}{\partial x^2}$ in $\mathbf{R}^2 \times \mathbf{R}_+$, the semi-implicit discretisation with the Neumann boundary condition

$$\frac{\mathbf{f}^{(k+1)} - \mathbf{f}^{(k)}}{\tau} = \frac{1}{2} \mathbf{D} \mathbf{f}^{(k+1)},$$

and the eigenvalue decomposition of the matrix \mathbf{D} yield the iteration form [20, 21]

$$\mathbf{f}^{(k+1)} = \Phi \left(\mathbf{I} - \frac{\tau}{2} \Lambda \right)^{-k} \Phi^\top \mathbf{f}, \quad \Lambda = ((\lambda_i \delta_{ij})),$$

where $\lambda_0 > \lambda_2 > \dots > \lambda_{n-1}$, for $\mathbf{f} = (f_0, f_1, \dots, f_{n-1})^\top$. This iteration form implies that the discrete scale transform is a linear transform from $L\{\varphi_i\}_{i=0}^{n-1}$ to $L\{\varphi_i\}_{i=0}^{n-1}$.

References

1. Frank, J. (ed.): *Electron Tomography: Methods for Three-Dimensional Visualization of Structures in the Cell*, 2nd edn. Springer, New York (2006). <https://doi.org/10.1007/978-0-387-69008-7>
2. Nguyen, H.T., Nguyen, L.-T.: The Laplacian pyramid with rational scaling factors and application on image denoising. In: 10th ISSPA, pp. 468–471 (2010)
3. Burt, P.J., Adelson, E.H.: The Laplacian pyramid as a compact image code. *IEEE Trans. Commun.* **31**, 532–540 (1983)
4. Burt, P.J., Adelson, E.H.: A multiresolution spline with application to image mosaics. *ACM Trans. Graph.* **2**, 217–236 (1983)
5. Thevenaz, P., Unser, M.: Optimization of mutual information for multiresolution image registration. *IEEE Trans. Image Process.* **9**, 2083–2099 (2000)
6. Ohnishi, N., Kameda, Y., Imiya, A., Dorst, L., Klette, R.: Dynamic multiresolution optical flow computation. In: Sommer, G., Klette, R. (eds.) *RobVis 2008*. LNCS, vol. 4931, pp. 1–15. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78157-8_1
7. Kropatsch, W.G.: A pyramid that grows by powers of 2. *Pattern Recogn. Lett.* **3**, 315–322 (1985)
8. Fletcher, P., Lu, C., Pizer, S.M., Joshi, S.: Principal geodesic analysis for the study of nonlinear statistics of shape. *IEEE TMD* **23**, 995–1005 (2004)
9. Henn, S., Witsch, K.: Multimodal image registration using a variational approach. *SIAM J. Sci. Comput.* **25**, 1429–1447 (2004)
10. Hermosillo, G., Chéfd’Hotel, C., Faugeras, O.: Variational methods for multimodal image matching. *IJCV* **50**, 329–343 (2002)
11. Hermosillo, G., Faugeras, O.: Well-posedness of two nonridged multimodal image registration methods. *SIAM J. Appl. Math.* **64**, 1550–1587 (2002)
12. Durrleman, S., Pennec, X., Trounev, A., Gerig, G., Ayache, N.: Spatiotemporal atlas estimation for developmental delay detection in longitudinal datasets. In: Yang, G.-Z., Hawkes, D., Rueckert, D., Noble, A., Taylor, C. (eds.) *MICCAI 2009*. LNCS, vol. 5761, pp. 297–304. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04268-3_37
13. Shepp, L.A., Kruskal, J.: Computerized tomography: the new medical X-ray technology. *Amer. Math. Monthly* **85**, 420–439 (1978)
14. Rumpf, M., Wirth, B.: A nonlinear elastic shape averaging approach. *SIAM J. Imaging Sci.* **2**, 800–833 (2009)
15. Inagaki, S., Itoh, H., Imiya, A.: Multiple alignment of spatiotemporal deformable objects for the average-organ computation. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) *ECCV 2014*. LNCS, vol. 8928, pp. 353–366. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16220-1_25
16. Ballester-Ripoll, R., Steiner, D., Pajarola, R.: Multiresolution volume filtering in the tensor compressed domain. *IEEE Trans. Visual Comput. Graphics* **24**, 2714–2727 (2018)
17. Ballester-Ripoll, R., Paredes, E.G., Pajarola, R.: Sobol tensor trains for global sensitivity analysis. *Reliab. Eng. Syst. Saf.* **183**, 311–322 (2019)

18. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. CoRR abs/1511.07122 (2015)
19. <http://brainweb.bic.mni.mcgill.ca/brainweb/>
20. Strang, G.: Computational Science and Engineering. Wellesley-Cambridge Press, Wellesley (2007)
21. Demmel, J.W.: Applied Numerical Linear Algebra. SIAM, Philadelphia (1997)
22. Fischer, B., Modersitzki, J.: Ill-posed medicine – an introduction to image registration. *Inverse Prob.* **24**, 1–17 (2008)
23. Modersitzki, J.: Numerical Methods for Image Registration. OUP, Oxford (2004)
24. Hermosillo, G., Chef d’Hotel, C., Faugeras, O.: Variational methods for multimodal image matching. *IJCV* **50**, 329–343 (2002)
25. Maes, F., Collignon, A., Vandermeulen, D., Marchal, G., Suetens, P.: Multimodality image registration by maximization of mutual information. *IEEE TMI* **16**, 187–198 (1997)

Multi-Sensor for Action and Gesture Recognition



Learning Spatiotemporal Representation Based on 3D Autoencoder for Anomaly Detection

Yunpeng Chang, Zhigang Tu^(✉), Bin Luo, and Qianqing Qin

Wuhan University, Wuhan 430079, China
{tuzhigang, luob}@whu.edu.cn

Abstract. Because of ambiguous definition of anomaly and the complexity of real data, anomaly detection in videos is of utmost importance in intelligent video surveillance. We approach this problem by learning a novel 3D convolution autoencoder architecture to capture informative spatiotemporal representation, and an 2D convolutional autoencoder to learn the pixel-wise correspondences of appearance and motion information to boost the performance. Experiments on some publicly available datasets demonstrate the effectiveness and competitive performance of our method on anomaly detection in videos.

Keywords: Anomaly detection · 3D convolution autoencoder · Spatiotemporal irregularity

1 Introduction

Anomaly detection in videos refers to the identification of events that do not conform to expected behavior. It is an important task in video analytics, e.g., it plays a crucial role in video surveillance. However, anomaly detection is an unsolved challenging issue due to the following problems: firstly, the realistic data is complex, anomaly data points may lie closely to the boundary of normal regions, e.g. skateboarders and walking people appear similarly in the application of camera surveillance, where skateboarders are anomaly objects and prohibited in pedestrian footpaths. Secondly, the labelled effective data is limited. Specially, the normal patterns are usually available or easy to be collected, but the abnormal samples are relatively few or costly.

In order to deal with these problems, several methods, which based on autoencoder for abnormality detection focus on modeling only the normal pattern of the videos [16], are proposed [18]. The main idea of this paradigm lies in the fact that only normal samples are needed at training time, while the detection of anomaly is according to measuring the distance from the learned normal pattern. Due to learning deviation in videos is very challenging as the definition is ill-defined [6], while learning ordinaries is relatively easier, this paradigm focuses on learning

Supported by Wuhan University.

the characteristics of regular temporal patterns with a very limited form of labeling - as it assumes that all events in the training videos are part of the regular patterns. To handle the issue of limited labelled data, [4] has been proposed to learn temporal regularity using 2D convolutional autoencoder. These methods just pay attention to temporal regularity implied by reconstruction error of video clips (Fig. 1).

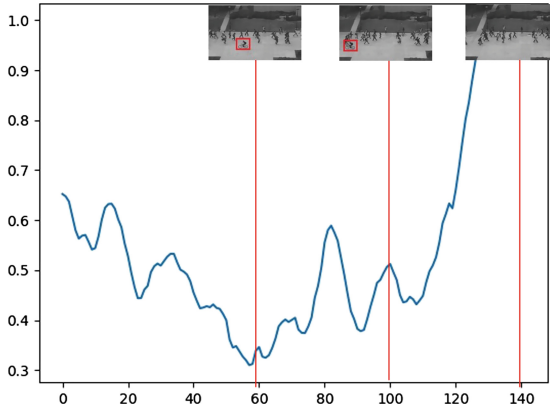


Fig. 1. Regularity score of a video sequence from Ped2 [8] and the red box locate the ground truth of abnormal events. When some abnormal events occurred, e.g., a bicycle intrudes, the regularity score drops significantly, and when there are no irregular motion, the regularity score is relatively high. (Color figure online)

In this work, we design a novel 3D convolution autoencoder architecture to learn the pixel-wise correspondences of appearance and motion information to capture informative spatiotemporal representation. In addition, our architecture can efficiently and effectively learn some semantics representation which are useful for less supervised video tasks. Since abnormal events are usually less than normal events in videos, it will be efficient if we formulate video anomaly detection as a one-class problem whose outliers are the anomaly.

In brief, our approach considers both appearance and motion features based on the perception that compared with normal behaviors, an abnormal behavior differs in their appearance characteristic or motion properties or both. In summary, this paper makes the following contributions:

- We propose a novel 3D autoencoder architecture which has temporal dimension to capture informative spatiotemporal representation to detect anomaly in videos by leveraging only weakly labeled videos end-to-end.
- We exploit to effectively capture both motion and appearance descriptors and appropriately combine them by an additional 2D convolutional autoencoder to learn the pixel-wise correspondences of them to boost the performance.

2 Related Work

2.1 Anomaly Detection Based on Hand-Crafted Features

Early work usually utilizes low-level trajectory features, a sequence of image coordinates, to represent the regular patterns [16]. However, these methods are not robust in complex or crowded scenes that with challenging cases, e.g., occlusions and shadows, because trajectory features are based on object tracking and it is very easy to fail in these conditions. Taking consideration of the shortcomings of trajectory, more useful low-level spatial-temporal features, such as histogram of oriented gradients (HOG) [1], histogram of oriented flows (HOF) [2] are exploited.

2.2 Anomaly Detection with Weak Supervision

Most video based anomaly detection approaches involve a local feature extraction step followed by learning a model on training video. [3, 11] use temporal coherency prior on adjacent frames to train an autoencoder network. [14] introduces label-free supervision which uses constraint learning combined with physics and domain knowledge to solve three computer vision tasks including tracking objects and a walking man. Recurrent Neural Network (RNN) and its long short term memory (LSTM) variant have been widely used for sequential data modeling, [13] utilized encoder LSTM to extract features and uses decoder LSTMs to the task of reconstruction tasks.

2.3 Autoencoder

Autoencoder is first applied to reduce dimensionality. [5] proposed a deep autoencoder initialized by RBMs. Then, to extract features more robustly, other variants of autoencoder are presented. Sparse autoencoder [10], denoising autoencoder [17], contractive autoencoder [12]. Convolutional autoencoder has been presented in [9]. The authors consider the 2D image as input and construct stacked convolutional autoencoders for initializing CNNs. [4] applied 2D convolutional autoencoder to learn temporal regularity. Based on these work, we proposed a novel 3D autoencoder architecture for video anomaly detection with the usage of both appearance and motion.

3 Approach

Supervised learning has achieved good performance on some video tasks, e.g., video recognition and action detection, however, it is difficult to apply supervised learning methods to the application of anomaly detection due to the lack of sufficient labeled abnormal events. To tackle those difficulties, we use 3D convolutional autoencoders to learn regularity in video sequences. The intuition is that the learned autoencoders is able to reconstruct the motion signatures presented in regular videos with low error but unable to accurately reconstruct motions in irregular videos. In other words, the autoencoder can model the complex distribution of the regular dynamics of appearance changes.

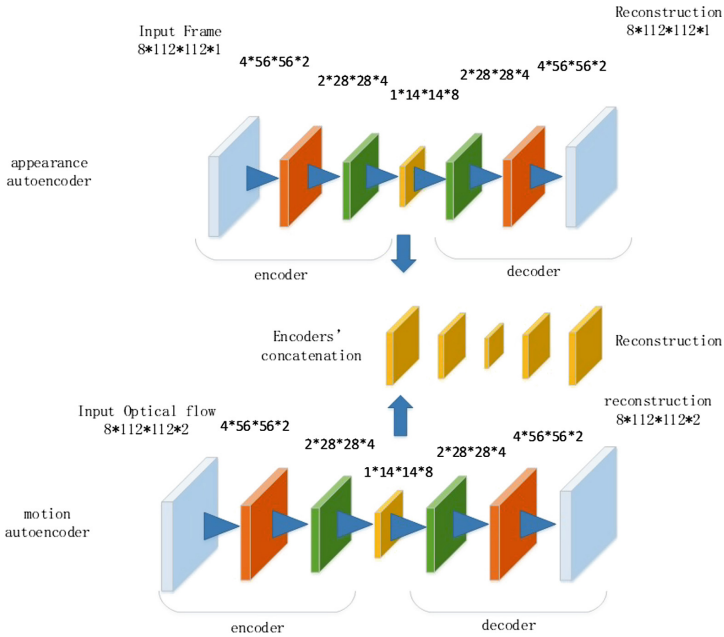


Fig. 2. Overview of our video anomaly detection architectures framework

3.1 Architectures

The proposed approach for detecting anomalous events rely on three autoencoders networks associated to different inputs to learn appearance, motion features, and a joint representation of them. The basic architecture of the proposed autoencoder network is illustrated in Fig. 2.

For the encoder part, we choose a shallow network with less than 4 layers. Then, the number of neurons is reduced by half in the next layer until reaching the “bottleneck” hidden layer. The decoder part has a symmetric structure with respect to the encoder part. To learn spatiotemporal features, 3D convolutional networks are introduced by [15]. Different from 2D convolution, 3D convolution has one more dimension: temporal. The output of one kernel convolves feature maps is 3-dimensional. 3D deconvolution is used in our decoder. It’s an inverse operation of 3D convolution. Deconvolution is also called fractionally strided convolution or transposed convolution.

Appearance Autoencoder. The first 3D-autoencoder learns mid-level appearance representations from the original image. We take short video clips in a temporal sliding windows to capture L consecutive frames as the input. The

objective function of appearance autoencoder is defined as:

$$L_{appearance} = \sum_{i,j}^{w,h} (x_{i,j} - \hat{x}_{i,j})^2 \quad (1)$$

Where $x_{i,j}$ represents the pixel intensity of input frame, and $\hat{x}_{i,j}$ represents the intensity of reconstruction result according to appearance autoencoder.

Motion Autoencoder. The second autoencoder is used to learn the motion features. Dense optical flow is computed to represent the motion. A dense optical flow can be seen as a set of displacement vector fields between the pairs of two consecutive frames t and $t+1$. Formally, optical flow consists of x and y displacement vectors for every position in the frame which indicate the horizontal and vertical movement components. We stack the optical flow of L consecutive frames to create $2L$ input. The objective function of motion autoencoder is expressed as:

$$L_{motion} = \sum_{i,j}^{w,h} (y_{i,j} - \hat{y}_{i,j})^2 \quad (2)$$

Where $y_{i,j}$ represents the optical flow value of input frames, and $\hat{y}_{i,j}$ represents the reconstruction result of the motion autoencoder.

Fusion Autoencoder. In contrast to the traditional methods [Reference] only extract appearance and motion features, to learn the pixel-wise correspondences between spatial and temporal features, we propose to fuse these two features to learn a joint representation by a 2D autoencoder:

$$L_{fusion} = \sum_{i,j}^{w,h} (z_{i,j} - \hat{z}_{i,j})^2 \quad (3)$$

We concatenate the output of appearance encoder and motion encoder to stack the two feature maps at the same spatial locations across the feature channels, and put the concatenated feature maps to a 2D autoencoders.

3.2 Objective Function

Our architecture’s objective function consists of the Euclidean loss between the input feature and the reconstructed feature of the three components:

$$L_{total} = L_{appearance} \times \lambda_{appearance} + L_{motion} \times \lambda_{motion} + L_{fusion} \times \lambda_{fusion} \quad (4)$$

To train the network, the intensity of frame pixels and optical flow in all frames are normalized to $[0,1]$. For different datasets, the coefficient factors $\lambda_{appearance}$, λ_{motion} , and λ_{fusion} can be manually setted. In our experiments, we set $\lambda_{appearance}$, λ_{motion} , and λ_{fusion} as 1.0, 0.1 and 10.0 respectively.

3.3 Regularity Score

Following [4], given the reconstruction error of a frame and optical flow which are obtained by summing up all the pixel-wise errors, we compute the regularity score of a frame t as follow:

$$g(t) = 1 - \frac{L(t) - \min_t(L(t))}{\max_t(L(t))} \quad (5)$$

4 Experiment

In this section, we evaluate our proposed method by testing the effects of its different components and comparing it with state of the arts on three publicly available anomaly detection datasets, including the CUHK Avenue dataset, the UCSD Pedestrian dataset and the ShanghaiTech dataset. Note that our model is not fine-tuned to one dataset. It is general enough to capture regularities across multiple datasets.

4.1 Datasets

We train our model based on three datasets: UCSD pedestrian [8] and Avenue [7], and the ShanghaiTech dataset.

The UCSD dataset contains two parts: The UCSD Pedestrian 1 (Ped1) dataset and the UCSD Pedestrian 2 (Ped2) dataset. The UCSD Pedestrian 1 (Ped1) dataset includes 34 training videos and 36 testing ones with 40 irregular events. All of these abnormal cases are about vehicles such as bicycles and cars. The UCSD Pedestrian 2 (Ped2) dataset contains 16 training videos and 12 testing videos with 12 abnormal events. The definition of anomaly for Ped2 is the same with Ped1. Usually different methods are evaluated on these two parts separately.

Avenue dataset is a static camera dataset in front of a subway station. 12 training video samples only contain normal videos, and 16 testing video samples are composed of both normal and abnormal video events. Each frame in this dataset has 640×320 pixels.

The ShanghaiTech dataset contains 330 training videos and 107 testing ones with 130 abnormal events. Totally, it consists of 13 scenes and various anomaly types.

4.2 Evaluation Metric

In the literature of anomaly detection [7,8], a popular evaluation metric is to calculate the Receiver Operation Characteristic (ROC) by gradually changing the threshold of regular scores. Then the Area Under Curve (AUC) is cumulated to a scalar for performance evaluation. A higher value indicates better anomaly detection performance. In this paper, we use frame-level AUC for performance evaluation.

4.3 Anomalous Event Detection

As our model learns the temporal regularity, it can be used for detecting anomalous events in a weakly supervised manner. We use our proposed autoencoders' objective function and regularity score to analyze global detection in frame level. Table 1 compares the anomaly detection accuracies of our autoencoders against state-of-the-art methods. We can see our method performs competitively to these methods. Figure 3 shows part of UCSD Ped2 dataset results' regularity scores as a function of frame number.

Table 1. AUC of different methods on the Ped1, Ped2, Avenue and ShanghaiTech datasets.

Algorithm	UCSD Ped1	UCSD Ped2	Avenue	ShanghaiTech
Conve-AE	75.00%	85.00%	80.00%	–
ConvLSTM-AE	75.50%	88.10	77.00%	–
Stacked RNN	–	92.20%	81.70%	68.00%
Our method	82.55%	90.95%	83.25%	71.52%

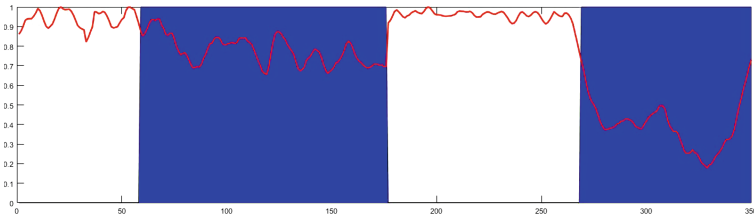


Fig. 3. Parts of temporal regularity score of UCSDPed2. The regularity score imply the possibility of normal, and blue shaded regions are the anomaly in groundtruth (Color figure online)

5 Conclusions

For video anomaly detection task, we proposed a 3D autoencoder architecture to learn spatiotemporal irregularity. We first designed an effective 3D convolutional autoencoder to extract informative representation of spatial and temporal information. Then, to learn the pixel-wise correspondences of them, we build an additional 2D convolutional autoencoder to fuse the representations of the appearance and motion to take advantages of their complementary. Our model is generalizable across multiple datasets, and quantitative analysis on three datasets shows that our method performs competitively to state-of-art methods.

Acknowledgment. The work is supported by the funding CFW-18-413100063 of Wuhan University. It is also supported by the Huawei-Wuhan University Funding (No. 250000916) and the National Key Research and Development Program of China (No. 2018YFB1600600).










References

1. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 886–893. IEEE (2005)
2. Dalal, N., Triggs, B., Schmid, C.: Human detection using oriented histograms of flow and appearance. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 428–441. Springer, Heidelberg (2006). https://doi.org/10.1007/11744047_33
3. Goroshin, R., Bruna, J., Tompson, J., Eigen, D., LeCun, Y.: Unsupervised feature learning from temporal data. arXiv preprint [arXiv:1504.02518](https://arxiv.org/abs/1504.02518) (2015)
4. Hasan, M., Choi, J., Neumann, J., Roy-Chowdhury, A.K., Davis, L.S.: Learning temporal regularity in video sequences. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 733–742 (2016)
5. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. *Science* **313**(5786), 504–507 (2006)
6. Li, Y., Liu, W., Huang, Q.: Traffic anomaly detection based on image descriptor in videos. *Multimed. Tools Appl.* **75**(5), 2487–2505 (2016)
7. Lu, C., Shi, J., Jia, J.: Abnormal event detection at 150 FPS in MATLAB. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2720–2727 (2013)
8. Mahadevan, V., Li, W., Bhalodia, V., Vasconcelos, N.: Anomaly detection in crowded scenes. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1975–1981. IEEE (2010)
9. Masci, J., Meier, U., Cireşan, D., Schmidhuber, J.: Stacked convolutional auto-encoders for hierarchical feature extraction. In: Honkela, T., Duch, W., Girolami, M., Kaski, S. (eds.) ICANN 2011. LNCS, vol. 6791, pp. 52–59. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21735-7_7
10. Poultney, C., Chopra, S., Cun, Y.L., et al.: Efficient learning of sparse representations with an energy-based model. In: Advances in Neural Information Processing Systems, pp. 1137–1144 (2007)
11. Ramanathan, V., Tang, K., Mori, G., Fei-Fei, L.: Learning temporal embeddings for complex video analysis. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4471–4479 (2015)
12. Rifai, S., Vincent, P., Muller, X., Glorot, X., Bengio, Y.: Contractive auto-encoders: explicit invariance during feature extraction. In: Proceedings of the 28th International Conference on International Conference on Machine Learning, pp. 833–840. Omnipress (2011)
13. Srivastava, N., Mansimov, E., Salakhutdinov, R.: Unsupervised learning of video representations using LSTMs. In: International Conference on Machine Learning, pp. 843–852 (2015)
14. Stewart, R., Ermon, S.: Label-free supervision of neural networks with physics and domain knowledge. In: AAAI, vol. 1, pp. 1–7 (2017)

15. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4489–4497 (2015)
16. Tung, F., Zelek, J.S., Clausi, D.A.: Goal-based trajectory analysis for unusual behaviour detection in intelligent surveillance. *Image Vis. Comput.* **29**(4), 230–240 (2011)
17. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.-A.: Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th International Conference on Machine Learning, pp. 1096–1103. ACM (2008)
18. Dan, X., Yan, Y., Ricci, E., Sebe, N.: Detecting anomalous events in videos by learning deep representations of appearance and motion. *Comput. Vis. Image Underst.* **156**, 117–127 (2017)



Multi-view Discriminant Analysis for Dynamic Hand Gesture Recognition

Huong-Giang Doan¹ , Thanh-Hai Tran² , Hai Vu² , Thi-Lan Le² ,
Van-Toi Nguyen³ , Sang Viet Dinh⁴ , Thi-Oanh Nguyen⁴ ,
Thi-Thuy Nguyen⁵ , and Duy-Cuong Nguyen⁶ 

- ¹ Control and Automation Faculty, Electric Power University, Hanoi, Vietnam
giangdth@epu.edu.vn
- ² International Research Institute MICA,
Hanoi University of Science and Technology, Hanoi, Vietnam
{thanh-hai.tran,hai.vu,thi-lan.le}@mica.edu.vn
- ³ Telecommunications Institute of Technology, Ho Chi Minh City, Vietnam
ntvoicntt@gmail.com
- ⁴ Hanoi University of Science and Technology, Hanoi, Vietnam
{sangdv,oanhnt}@soict.hust.edu.vn
- ⁵ VietNam National University Agriculture, Hanoi, Vietnam
ntthuy@vnua.edu.vn
- ⁶ Panasonic R&D Center, Hanoi, Vietnam
nguyenduycuong2004@gmail.com

Abstract. Although there have been attempts to tackle the problem of hand gesture recognition “in-the-wild”, deployment of such methods in practical applications still face major issues such as view point change, clustered background and low resolution of hand regions. In this paper, we investigate these issues based on a frame-work that is intensively designed in terms of both varying features and multi-view analysis. In the framework, we embed both hand-crafted features and learnt features using Convolutional Neural Network (CNN) for gesture representation at single view. We then employ multi-view discriminant analysis (MvDA) based techniques to build a discriminant common space by jointly learning multiple view-specific linear transforms from multiple views. To evaluate the effectiveness of the proposed frame-work, we construct a new multi-view dataset of twelve gestures. These gestures are captured by five cameras uniformly spaced on the half of a circle frontally surrounding the user in the context of human machine interaction. The performance of each designed scheme in the proposed framework is then evaluated. We report accuracy and discuss the results in view of developing practical applications. Experimental results show promising performance for developing a natural and friendly hand-gesture based applications.

This material is based upon work supported by the Air Force Office of Scientific Research under award number FA2386-17-1-4056.

Electronic supplementary material The online version of this chapter (https://doi.org/10.1007/978-981-15-3651-9_18) contains supplementary material, which is available to authorized users.

Keywords: Multi-view learning · Cross-view recognition · Hand gesture recognition · Dynamic hand gesture dataset

1 Introduction

Hand gestures are becoming one of the most natural means for human machine interaction (HMI). In the last decades, numerous techniques for hand gesture recognition have been proposed and developed in practical applications, for example sign-language recognition [5]. Thanks to recent advances of the depth sensors (e.g., PrimeSense, Kinect) and Deep Neuron Networks (DNN), many hand-related works (e.g., hand gesture recognition, hand pose estimation) demonstrated impressive results [12, 13]. However, as denoted in a comprehensive and recent survey [13], the main challenges such as view-point change of active hands or cluttered background, low-resolution of hand regions are still remaining. Many existing methods will be failed given those challenges. In context of developing practical applications, like gaming interface, or home appliance controlling system [2], using hand gestures in a natural way is always required. An end-user should not directly point his/her hand to the sensor as well as would not care about a valid distance from the subject to the sensor in most of interactive situations. The majority of existing works deal with hand gestures recognition from a common viewpoint or in a specific context (e.g., a subject is sitting on a chair and point his/her hand to the sensor). Different viewpoints result in different hand poses, hand appearances and even background and/or lighting. This degrades dramatically the performance of pre-trained models. Therefore, proposing robust methods for recognizing hand gestures from unknown view-point (or we adopt the term from [13], “in-the-wild” hands) is pursued in this work.

Our focus in this paper is systematically investigating the performance of cross-view action recognition method on human hand gestures and analysing how to improve it. We propose a framework which takes both handcrafted features and learnt features using CNN for gesture representation at single view. Then we employ MvDA based techniques to build a discriminant common space by jointly learning multiple view-specific linear transforms from multiple views. To the best of our knowledge, this is the first intensive work to explore dynamic hand gestures under different viewpoints.

To this end, a crucial requirement is to build a dataset of dynamic hand gestures observed from different viewpoints. In fact, lack of existing multi-view datasets of hand gestures, recognizing hand gestures under different viewpoints could not be explored. Although a related topic such as hand pose estimation has been extensively studied recently with many public datasets at different viewpoints [13]. Unfortunately, dynamic hand gestures datasets with a large number of gesture types and varying view-points are still not available. In [9], the authors have introduced a dataset of five hand gestures, captured by five cameras at different viewpoints. In [16], a dataset of two gestures taken from four Kinects has been collected for investigating the role of multiple views in

authentication problem. In those works, the number of gestures are very limited (two or five gestures) and not designed to hand gestures but for body gestures in general. As indicated in [11], most of hand gestures datasets are collected by single camera and they introduced a new multi-view dataset of eight gestures taken at different locations by two cameras (side and frontal views). In this paper, we introduce a set of twelve gestures captured with five camera. This gesture set allows mapping to more commands for controlling equipment in reality. Due to the higher number of gestures, gesture recognition will be more challenging.

Thanks to the proposed frame-work and the constructed dataset, performances of the gestures recognition from different views are deeply investigated. Consequently, developing a practical application is feasible. Wherein, a “sensitive area” can be used to measure the accuracy/sensitivity of a gesture-based control from ambiguous directions. In that context, the end-user could stand in any position and orientation in the room while doing a control gesture. She/he could have a habit of looking forward to the equipment to be controlled but not the device to capture the image. The constructed dataset is made publicly available.

2 Related Work

In the literature, there are many intensive surveys on the hand gesture recognition [21], particularly, recent hand pose estimation [13]. However, to our best knowledge, the related works on hand gestures recognition from ambiguous viewpoints are still very limited. Although dynamic hand gesture could belong to human actions, where general methods for action recognition could be directly applied. However, deploying such related techniques are prevented from specific characteristics of hand gestures, such as hand region has small resolution but high DoF, the fingers are easily self-occluded, various temporal factors/noises (frame-rate, speed of hand movement, phase variation). In following sections, we divide works into three topics related to the proposed method.

2.1 Hand Gestures for Controlling Home Appliances

Nowadays, dynamic hand gesture-based controls are developed in a wide range of common home appliances such as Television, air-conditions, fan, light, door. For instance, a Samsung smart TV now consists of a de-factor function providing a hand gesture-based controlled by moving one/two hands. Works in [22] attempts using static hand gesture system by a dynamic gesture. Their system is capable of rejecting unintentional gestures thanks to the start and stop routines. [8] utilized three types of sensors with six dynamic hand gestures conveying commands to control a television. [19] used two USB pan-tilt cameras. The system strongly depends on the assumption that a gesture is performed if the hand is moved in a high speed. Although a series of the works has been listed, the recognition of hand gestures from ambiguous views have been not investigated yet. In context of developing a feasible application, we point out that a learnt model from a

certain view can be failed from different views. To develop a natural application, without intending control direction of the end-user is required.

2.2 Multiple-View Learning

Data in our real-world is always in multiple views and/or multi-modalities. In one hand, multi-view learning has been known as machine learning algorithms which consider learning data from different views at the same time in order to improve the system performance. In the other hand, multiple view learning is also known as data fusion or data integration from multiple features. In [17], multi-view learning algorithms could be considered in three aspects: co-training; multiple kernel learning; and subspace learning. In [1], authors proposed the large-margin framework for multi-view data that is based on an un-directed latent space Markov network to name a few. Works in [20] introduces multiple view-specific projection matrices and project a recognition target from a certain view by a corresponding view-specific projection matrix into a common discriminant subspace.

2.3 Viewpoint-Invariant Recognition

Viewpoint variations often make action recognition challenging because the same actions can be seen different from different views. Many view-invariant approaches have been proposed. [4] proposes a view-invariant matching method based on epipolar geometry between actor silhouettes without tracking and explicit point correspondences. [7] learns two view-specific transformations for the source and target views, and then generated a sequence of linear transformations of action descriptors as the virtual views to connect two views. [18] introduces a 4D view-invariant action feature extraction to encode the shape and motion information of actors observed from multiple views. These approaches lead to computationally intensive algorithms because they must find the best match between 3D and 2D observations over a large model parameter space.

3 Methods for Gesture Recognition

Our proposed framework is illustrated in Fig. 1. It consists of two main blocks: The first block extracts features from each video sample captured from each single view. We call these features *private features*. The second one jointly learns a set of transformations to project all private features into a common space to generate viewpoint-invariant representation of gestures. Then any classifier can be used to classify these gestures in the common space. In the following, we will present each technique in detail.

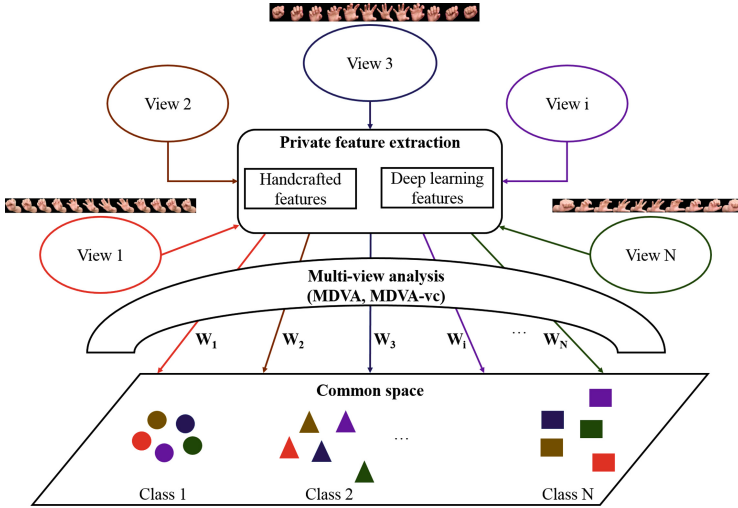


Fig. 1. Proposed framework for dynamic hand gesture recognition

3.1 Private Features Extraction from Single View

To extract private features for representing a gesture, we adopt two techniques: one extracts hand-crafted features and the other learns features using convolutional neural network. We would like to investigate which features could deal better with viewpoint change in case of dynamic hand gestures.

Extracting Hand-Crafted Features. Hand-crafted features for action representation usually take both spatial and temporal cues of the action into account. In this work, hand-crafted features are extracted rely on techniques that is presented detail in our previous research [3], due to its efficiency and its out-performance on gesture recognition problem. That uses ISOMAP as non-linear dimension reduction for representing spatial cue and Kanade Lucas Tomasi feature tracker (KLT) to capture temporal cue.

Spatial Features Extraction. We construct a low-dimension space of hand postures from each frame of gesture sequences by utilizing a manifold learning technique. Suppose that at the i^{th} view, given a set of N_i bounding boxes of postures $\mathbf{P}^{(i)} = \{P_1^i, \dots, P_{N_i}^i\}$. These postures can be achieved using any automatic hand detection and/or segmentation technique. In this paper, to avoid impact of such techniques to the conclusion of the studied framework, we manually annotate hand region in each frame (see Fig. 5) and utilise segmented hand regions to input to our algorithm. Each posture $P_k^i, k = (1, \dots, N_i)$ could have different size. We normalize all of them to the same size and reshaping each to a row vector. Then we employ conventional non-linear dimension reduction technique ISOMAP [14] to compute the corresponding coordinate vectors

$\mathbf{Y}^{(i)} = \{Y_k^i \in R^d, k = (1, \dots, N_i)\}$ in the d -dimensional manifold space ($d \ll D$), where D is dimension of original data. The value of d could be empirically chosen. We use residual variance R_d to evaluate the error of dimensional reduction and select three first components ($d = 3$) in the manifold space to extract spatial features of each hand shape/posture. A hand gesture with M -frames length at the i^{th} view is now represented in the manifold as follows:

$$Y_M^{iG} = \{(Y_{1,1}^i, Y_{1,2}^i, Y_{1,3}^i), (Y_{2,1}^i, Y_{2,2}^i, Y_{2,3}^i), \dots, (Y_{M,1}^i, Y_{M,2}^i, Y_{M,3}^i)\} \quad (1)$$

Temporal Features Extraction. We use KLT (Kanade-Lucas-Tomasi) tracker [10] to extract hand movement trajectory. Then we extract temporal features through three main steps: (1) connect tracked points to create trajectories; (2) select the most significant trajectories; (3) compute average trajectories of these significant trajectories (x, y) . The most significant trajectory is considered as the longest ones among all. The average trajectory represents the main direction of hand movement. The temporal features (Tr_M^{iG}) extracted from M -frames sequence of gesture at the i^{th} view as:

$$Tr_M^{iG} = \{(x_1^i, y_1^i), (x_2^i, y_2^i), \dots, (x_M^i, y_M^i)\} \quad (2)$$

Phase Alignment: Once the spatial and temporal features are extracted, they are combined to completely represent dynamic hand gestures $\{Y_M^{iG}, Tr_M^{iG}\}$ at the i^{th} view. In a practical situation, the subject can perform the same gesture differently at different times and this difference is more remarkable among subjects. In that way, the length of the gestures is different from each other. We consider it as *phase variation* problem. To overcome this, we apply also the interpolation scheme as described in our previous work [3] so that the hand gesture sequences have the same length while maximizing inter-period phase continuity.

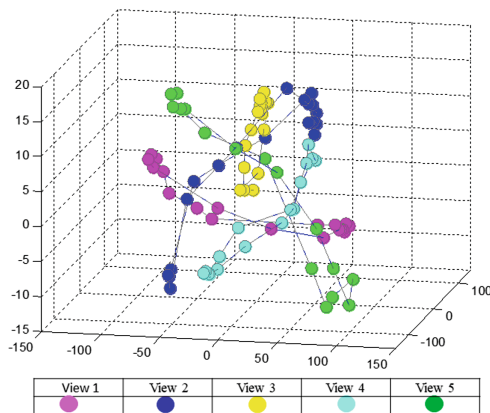


Fig. 2. Illustration of a gesture belonging to the 6th class in private space from different viewpoints.

Figure 2 illustrates a gesture sample belonging to the 6th class in the new private feature space. In this figure, we use the first two elements of the temporal features and the first element of spatial features. Different colors correspond to different views. Obviously, private features in different views do not converge. As a result, the cross-view recognition will be very challenging.

Feature Extraction Using 3D Convolutional Neural Network. CNN is a must-try technique in any scheme of recognition. Recently, the 3D convolutional neural network (C3D) has shown to be very efficient for human action recognition (as presented in our previous research [15]). In this paper, it will be utilized as a private feature extractor of dynamic gestures. C3D composes of 8 convolutional layers, 5 max pooling and 2 fully connected layers followed by a soft-max output layer. In this network, the convolutional operation is 3D convolution which aims to capture both spatial and temporal information of video.

Originally, C3D was trained on human action dataset Sport 1M. However, to adapt to use for hand gestures, we have fine tuned the network on our dataset of hand gestures. We then use the fine-tuned network to extract features of 4096 dimensions at FC6 layer. As our dataset is still small in terms of number of samples for training (about 216 samples per view), we have applied data augmentation and utilize zero padding and this results in a better performance compared to using the original dataset for training.

3.2 Learning View-Invariant Representation for Cross-View Recognition

As mentioned previously, private features of the same gesture are very different at different viewpoints. They should be represented in another common space to be converged. There exists a number of techniques to build the viewpoint invariant representation. In this paper, we will deploy a variant of multi-view discriminant analysis methods: Multi-view discriminant analysis (MvDA) and Multi-view discriminant analysis with view consistency (MvDA-vc). These techniques have been used for view-invariant face recognition [6]. However, most of multi-view discriminant analysis in the literature as well as in [6] were exploited for still images. To the best of our knowledge, our work is the first one to build common space for video sequences. We will see how such techniques could help to improve cross-view recognition.

Multi-view Discriminant Analysis (MvDA). Suppose that gestures belonging to c classes are observed from v views, the number of samples from the j^{th} view of the i^{th} class is n_{ij} . We define $X = \{\mathbf{x}_{ijk} | i = (1, \dots, c); j = (1, \dots, v); k = (1, \dots, n_{ij})\}$ as samples from v views where $\mathbf{x}_{ijk} \in R^{d_j}$ is the k^{th} sample from the j^{th} view of the i^{th} class, d_j is the dimensions of data at the j^{th} view. Here \mathbf{x}_{ijk} can be a handcrafted feature vector or a learnt feature vector extracted using C3D. The multi-view discriminant analysis method tries to determine a set of v linear transformations to project all gesture samples from

each view $j = (1, \dots, v)$ to a common space. The projection results of X on the common space is denoted by $Y = \{\mathbf{y}_{ijk} = \mathbf{w}_j^T \mathbf{x}_{ijk} | i = (1, \dots, c); j = (1, \dots, v); k = (1, \dots, n_{ij})\}$. The common space is built by maximizing the between-class variation \mathbf{S}_B^y while minimizing the within-class variation \mathbf{S}_W^y from all views. \mathbf{S}_B^y and \mathbf{S}_W^y are computed as follows:

$$\mathbf{S}_W^y = \sum_{i=1}^c \sum_{j=1}^v \sum_{k=1}^{n_{ij}} (y_{ijk} - \mu_i)(y_{ijk} - \mu_i)^T \quad (3)$$

$$\mathbf{S}_B^y = \sum_{i=1}^c n_i (\mu_i - \mu)(\mu_i - \mu)^T \quad (4)$$

where $\mu_i = \frac{1}{n_i} \sum_{j=1}^v \sum_{k=1}^{n_{ij}} \mathbf{y}_{ijk}$ is the mean of all samples of the i^{th} class from all views in the common space; $\mu = \frac{1}{n} \sum_{i=1}^c \sum_{j=1}^v \sum_{k=1}^{n_{ij}} \mathbf{y}_{ijk}$ is the mean of all samples of all classes from all views in the common space; $n = \sum_{i=1}^c n_i$.

Then the objective is formulated by a Reyleigh quotient:

$$(\mathbf{w}_1^*, \mathbf{w}_2^*, \dots, \mathbf{w}_v^*) = \arg \max_{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_v} \frac{Tr(\mathbf{S}_B^y)}{Tr(\mathbf{S}_W^y)} \quad (5)$$

According to [6], the optimization problem could be analytically solved through generalized eigenvalue decomposition.

Multi-view Discriminant Analysis with View Consistency (MvDA-vc).

In [6], the authors observed that as multiple views correspond to the same objects, there should be some correspondence between multiple views, that means if $\mathbf{X}_1, \mathbf{X}_2$ are observed at two views v_1, v_2 , then there exists a certain transformation \mathbf{R} so that $\mathbf{X}_1 = \mathbf{R}\mathbf{X}_2$. As a result, the transformations obtained from two views have the similar relationship: $\mathbf{w}_1 = \mathbf{R}\mathbf{w}_2$. Let us define β_i that captures the structure of the transformation \mathbf{w}_i . Then the β_1 and β_2 capturing the structures of two transformations of two views 1 and 2 should be similar: $\beta_1 = \beta_2$.

Generalizing to v views, suppose that $\beta_i, i = (1, \dots, v)$ captures the structures of v transformations \mathbf{w}_i . Following the above observation, the $\beta_i, i = (1, \dots, v)$ should resemble mutually. That means the similarity between the pair of β_i and β_j should be minimized.

$$\sum_{i,j=1}^v \|\beta_i - \beta_j\|_2^2 \quad (6)$$

This term is called in [6] *view consistency* and will be added to the denominator of Eq. (5)

$$(\mathbf{w}_1^*, \mathbf{w}_2^*, \dots, \mathbf{w}_v^*) = \arg \max_{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_v} \frac{Tr(\mathbf{S}_B^y)}{Tr(\mathbf{S}_W^y) + \alpha \sum_{i,j=1}^v \|\beta_i - \beta_j\|_2^2} \quad (7)$$

Similarly, this optimization problem could be analytically solved by relaxing to the ratio trace problem as Eq. (5). In the Eq. (7), α is an empirically chosen

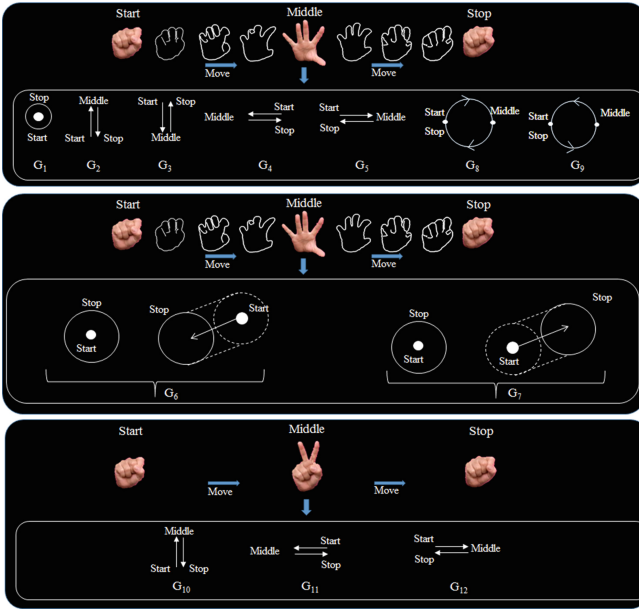


Fig. 3. Set of twelve dynamic hand gestures.

parameter. It puts a weight on the view-consistency assumption. When $\alpha = 0$, the MvDA-vc becomes the original MvDA.

Once the transformations (\mathbf{w}_1^* , \mathbf{w}_2^* , ..., \mathbf{w}_v^*) have been determined, the projection of data from separate original view to the common space is straightforward. We consider the features in this common space as view-invariant features and can apply any classifier to do the recognition. In this paper, we will simply use K-NN as classifier. The selection of the better classifier will be a future work.

4 Dataset and Experimental Results

4.1 Multiview Dataset

The evaluation of robustness of hand gesture recognition w.r.t viewpoint changes was not considered in the literature. Therefore, there does not exist a dataset dedicated to this problem. In our work, we carefully design a dataset which is collected from multiple camera viewpoints in indoor environment with complex background. Our dataset consists of twelve dynamic hand gestures which correspond to controlling commands of electronic home appliances. Each gesture is a combination of hand movement following a pre-defined direction and changing of hand shape in a natural manner. For each gesture, hand starts from one position with close posture, it opens gradually at half cycle of movement then closes gradually to end at the same position and posture. Figure 3 illustrates the movement of hand and changes of postures during gesture implementation.

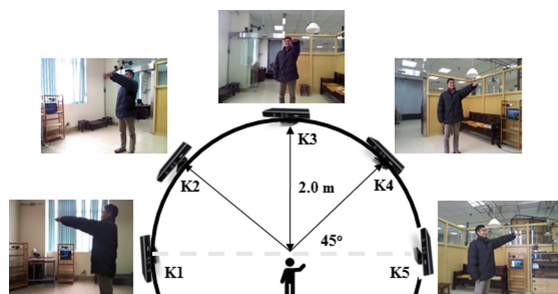


Fig. 4. Environment setup.



Fig. 5. Illustration of a gesture belonging to the 6th class observed from five different views.

Five Kinect sensors K1, K2, K3, K4, K5 are setup at five various positions in a simulation room of $4\text{ m} \times 4\text{ m}$ with a complex background (Fig. 4). This work aims to capture hand gestures under multiple viewpoints at the same time. Subjects are invited to stand at a nearly fixed position in front of five cameras at an approximate distance of 2 m. The Kinect sensor provides both RGB and Depth data. This setup allows to capture a multi-view and multi-modal dataset.

Twenty participants (13 males and 7 females) are voluntary to perform gestures one after another. Each subject performs one gesture three times. Totally, the dataset contains 900 ($5\text{ views} \times 12\text{ gestures} \times 5\text{ subjects} \times 3\text{ times}$) dynamic hand gestures. The frame rate is 20 fps and frame resolution is set to 640×480 . Each gesture's length varies from 50 to 120 frames (phase variation problem). This leads to a huge number of frames to be processed. This dataset will be soon publicly available for research purpose. We have annotated the dataset at gesture level. Manual annotation at pixel level for every frame is a very time-consuming task. At writing time of this paper, gestures from six subjects have been annotated at pixel levels. As a result, in this work, we will evaluate on a subset of the whole collected data. Besides, although the dataset is multi-modal (RGB, skeleton, Depth), in this work, we will evaluate on only RGB channel. Figure 5 illustrates a gesture belonging to the 6th class observed from five different views and its correspondence in private feature space (Fig. 2). We observe a big difference in appearance of hand postures at different views.

Table 1. Average accuracy % of single view gesture recognition using handcrafted and learnt features with and without multi-view discriminant analysis.

<i>Train Test</i>	1 1	2 2	3 3	4 4	5 5	Avr.
C3D+SVM [15]	66.2	75.9	86.1	75.0	67.6	74.1
C3D+KNN	85.2	87.5	87.5	88.4	94.9	88.7
C3D+MvDA+KNN	77.3	77.8	86.6	80.5	80.1	80.4
C3D+MvDA-vc+KNN	88.4	89.8	92.6	90.7	98.6	92.0
ISOMAP+KLT+SVM [3]	63.7	70.9	81.2	74.2	61.2	70.2
ISOMAP+KLT+KNN	67.0	78.9	78.2	76.9	90.1	78.2
ISOMAP+KLT+MvDA+KNN	61.4	64.0	59.1	57.4	83.5	65.0
ISOMAP+KLT+MvDA-vc+KNN	88.7	87.7	92.7	94.7	98.7	92.5

4.2 Experimental Results

To evaluate the proposed framework, we follow one-leave-subject out in all experiment settings to ensure independent subject training. Each time, we use samples performed by one subject for testing and all samples performed by remaining subjects for training. The average accuracy is computed to evaluate performance of each technique. We evaluate how the conventional methods themselves could deal with variation of viewpoints and what is the role of multi-view discriminant analysis. It notices that when using MvDA or MvDA-vc in the framework, for both single and cross-view evaluations, all samples from every viewpoint will be projected onto the common space. After that, single view recognition uses training and testing samples from the same view in that common space while cross-view recognition uses training and testing data from two different views. The details of all experiments will be described in the next subsections.

Evaluation of Single View Recognition. In this part, we present single view recognition results obtained with two groups of methods with and without using multi-view discriminant analysis. The first group uses 3D convolutional neural network to extract features while the second group uses handcrafted features. Table 1 shows recognition accuracy obtained with single view evaluation protocol. That means the training and testing views are the same.

Discussion on the Use of Handcrafted and Learnt Features: Firstly, all of the methods using C3D features give the performance similar or lightly better than the ones using handcrafted features (ISOMAP+KLT). The results could be explained by the fact that C3D is more robust for characterizing hand movement. In addition, due to the fact that the C3D features extractor takes the whole video sequence as input for learning the model, it will learn the contextual background that makes good effects for distinguishing the gestures. In addition, an advantage of C3D features is it does not need a pre-processing step such as

Table 2. Average accuracy (%) of cross-view gesture recognition using handcrafted and learnt features with and without multi-view discriminant analysis.

<i>Train/Test</i>	1 2	1 3	1 4	1 5	2 1	2 3	2 4	2 5	3 1	3 2	3 4	3 5	4 1	4 2	4 3	4 5	5 1	5 2	5 3	5 4	Avr.
C3D+SVM [15]	44.9	28.2	25.4	19.4	25.4	59.2	37.0	21.7	21.3	35.2	35.2	20.4	24.1	26.8	59.2	42.1	17.6	20.8	28.7	54.6	32.4
C3D+KNN	9.3	7.9	9.3	6.5	5.7	5.1	9.3	3.7	8.3	9.3	9.3	5.1	6.9	6.9	5.1	10.2	6.9	8.8	6.5	6.9	7.3
C3D+MvDA+KNN	68.5	67.1	68.1	67.6	69.4	67.1	69.9	67.1	68.1	68.9	67.6	67.6	64.8	64.8	64.3	65.3	70.8	73.1	72.2	71.3	68.2
C3D+MvDA-vc+KNN	74.1	72.7	72.2	70.4	68.5	65.7	67.6	67.1	70.4	89.8	67.6	67.1	70.8	71.8	70.4	69.9	74.1	75.0	73.1	74.1	71.6
ISOMAP+KLT+SVM [3]	46.3	32.1	26.4	24.3	48.2	54.7	31.8	26.9	36.9	44.9	58.6	45.7	30.3	34.7	57.7	48.0	24.9	26.0	37.2	53.0	39.4
ISOMAP+KLT+KNN	46.9	30.4	17.5	12.2	51.2	50.2	30.4	17.8	20.1	49.2	56.1	32.7	12.2	25.7	60.4	53.8	17.8	14.2	43.9	62.4	35.5
ISOMAP+KLT+MvDA+KNN	67.0	68.7	69.6	67.3	65.7	63.7	61.4	65.0	67.0	66.3	66.7	68.3	65.7	66.7	69.3	66.0	73.3	74.6	74.6	75.2	67.5
ISOMAP+KLT+MvDA-vc+KNN	71.8	70.6	70.6	71.6	69.6	72.9	70.9	73.2	72.6	72.9	72.6	72.6	75.2	72.9	71.6	74.5	80.8	71.2	78.5	80.8	77.7

hand segmentation. In case that hand segmentation is too challenging, learning features is an alternative choice. However, C3D is much more time consuming and requires huge memory load so it should be carefully considered to deploy in practical application. One idea is to lighten the network architecture when the number of classes is small.

Discussion on the Variation of Viewpoints: We first discuss the results obtained by handcrafted features and CNN features without using multi-view discriminant analysis. C3D+SVM gives the highest accuracy at the third view (the frontal view - K_3) (86.1%) and degrades gradually in the views at the left side and right side of the user. The same for ISOMAP+KLT+SVM which obtained the highest accuracy (8.12%) at the 3rd view. However, when C3D or ISOMAP+KLT combine with KNN ($K=1$), the highest accuracy achieved at the most right view (K_5 - 94.9% by C3D+KNN and 90.1% by ISOMAP+KLT+KNN). In general, KNN gives better average accuracy (88.7% by C3D+KNN vs 74.1% by C3D+SVM; 78.2% by ISOMAP+KLT+KNN vs. 70.2% by ISOMAP+KLT+SVM). In worst cases, the accuracy without multi-view discriminant analysis, is only more than 60%.

Discussion on the Impact of Multi-view Discriminant Analysis. We see in Table 1 that recognition accuracy has significantly increased when we apply multi-view discriminant analysis. With C3D features, MvDA helps to increase the average accuracy on all pair of views from 74.1% to 80.4%. It continues to increase to 92.0% thanks to taking *view consistency* into account. Concerning handcrafted features ISOMAP+KLT, the multi-view discriminant analysis does not help to improve the recognition accuracy on single view. It may be explained by the fact that the projected points on the new common space are more scattered. In that case, the K-NN could be a too simple classifier for distinguishing different classes. However, as we can see later, multi-view discriminant analysis helps to improve significantly cross-view recognition. In both cases of using learnt or handcrafted features (the 4th row and the 8th row of the Table 1), multi-view discriminant analysis with view consistency helps to increase recognition accuracy for all views (92.0% with C3D+MvDA-vc+KNN and 92.5% with ISOMAP+KLT+MvDA-vc+KNN).

Evaluation of Cross View Recognition

Discussion on the Use of Handcrafted and Learnt Features: In cross-view evaluation, handcrafted features give lightly better average accuracy than C3D features (39.4% vs. 32.4%). However, both feature types give very low recognition accuracy without using multi-view discriminant analysis (the second row and the fifth row in Table 2). It notices that in this paper, we report only the use of SVM as classifiers according the two original papers [3, 15]. We have tried also the use of KNN but the accuracy is very low.

Discussion on the Variation of Viewpoints: We observe that when the two consecutive views are considered (e.g. the first view close to the second one), the recognition accuracy is better than when two views are far. For instance, accuracy could achieve to 59.2% with C3D features with the pair of training, testing view (2,3) or (4,3). However, the accuracy is only 17.6% with C3D features when training and testing views are too far. The same situation happens with handcrafted features (the fifth row of the Table 2). This is explained that both features are not robust to a large change of viewpoints. Besides, the accuracy is not symmetric when training and testing views interchange.

Discussion on Impact of Multi-view Discriminant Analysis: When using multi-view discriminant analysis, for both types of features, the accuracy improves impressively (from 32.4% to 68.2% with MvDA and 71.6% with MvDA-vc in case of C3D features; from 39.4% to 67.5% with MvDA and 77.7% with MvDA-vc in case of ISOMAP+KLT features). It is very interesting to notice that accuracy at each pair of (training, testing) views is now quite consistent, independent of the fact that the training and testing are close or far. This is a power of multi-discriminant analysis that builds the common space and helps to find out an invariant representation of gestures.

Impact of View Consistency on Recognition. As discussed previously, view consistency has strong impact on the performance of recognition in both cases (single view and cross-view). In this subsection, we study the influence of value α in Eq. 7 on the recognition. Figure 6 shows the evolution of average accuracy when the value α increased. We found that the highest accuracy obtained with $\alpha = 0.01$ using C3D features and $\alpha = 0.05$ with ISOMAP+KLT features. When α is increased, the accuracy is reduced. It means we should not put a big weight on view consistency because it could reduce the role of within-class variation S_W^y from all views in Eq. (7).

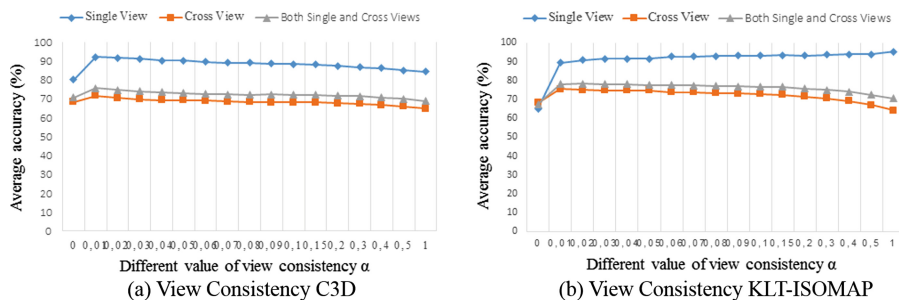


Fig. 6. Impact of view consistency term (α) on recognition performance using C3D and handcrafted (KLT+ISOMAP) features.

5 Conclusions

The paper proposes a framework to intensively study the impact of view change on recognition performance with both handcrafted features and learnt features. We conclude that the difference between training views and testing views could degrade strongly recognition performance. Multi-view discriminant analysis with view consistency assumption helps to boost cross-view recognition significantly. Even this, the average accuracy in cross-view evaluation is still low (the highest average accuracy is 77.7%) and needs to be improved. In the future, we plan to take benefit of multi-modalities; generate more multi-view data using Generative Adversarial Network to enrich the training set and test with the whole dataset. In addition, this paper also introduce a new multi-view and multi-modal dynamic hand gesture dataset in the context of human machine interaction. This dataset, publicly made available, will allow researchers in the community to deeply investigate new robust algorithms to be deployed in practical applications.

References

1. Chen, N., Zhu, J., Xing, E.P.: Predictive subspace learning for multi-view data: a large margin approach. In: Advances in Neural Information Processing Systems 23, pp. 361–369. Curran Associates, Inc. (2010)
2. Doan, H.G., Vu, H., Tran, T.H.: Recognition of hand gestures from cyclic hand movements using spatial-temporal features. In: Proceedings of the Sixth International Symposium on Information and Communication Technology, pp. 260–267. ACM (2015)
3. Doan, H.G., Vu, H., Tran, T.H.: Phase synchronization in a manifold space for recognizing dynamic hand gestures from periodic image sequence. In: 2016 IEEE RIVF International Conference on Computing & Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), pp. 163–168. IEEE (2016)
4. Gondal, I., Murshed, M., et al.: On dynamic scene geometry for view-invariant action matching. In: CVPR 2011, pp. 3305–3312 (2011)
5. Jangyodsuk, P., Conly, C., Athitsos, V.: Sign language recognition using dynamic time warping and hand shape distance based on histogram of oriented gradient

- features. In: Proceedings of the 7th International Conference on Pervasive Technologies Related to Assistive Environments, pp. 50:1–50:6 (2014)
6. Kan, M., Shan, S., Zhang, H., Lao, S., Chen, X.: Multi-view discriminant analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **38**(1), 188–194 (2016)
 7. Li, R., Zickler, T.: Discriminative virtual views for cross-view action recognition. In: 2012 IEEE Conference on CVPR, pp. 2855–2862
 8. Lian, S., Hu, W., Wang, K.: Automatic user state recognition for hand gesture based low-cost television control system. *IEEE Trans. Consum. Electron.* **60**(1), 107–115 (2014)
 9. Nguyen, D.H., Le, T.H., Tran, T.H., Vu, H., Le, T.L., Doan, H.G.: Hand segmentation under different viewpoints by combination of Mask R-CNN with tracking. In: 2018 5th Asian Conference on Defense Technology (ACDT), pp. 14–20 (2018)
 10. Shi, J., Tomasi, C.: Good features to track. In: 1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 593–600, June 1994
 11. Shukla, D., Erkent, Ö., Piater, J.: A multi-view hand gesture RGB-D dataset for human-robot interaction scenarios. In: 2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), pp. 1084–1091 (2016)
 12. Simon, T., Joo, H., Matthews, I., Sheikh, Y.: Hand keypoint detection in single images using multiview bootstrapping. In: International Conference on CVPR (2017)
 13. Supancic, J.S., Rogez, G., Yang, Y., Shotton, J., Ramanan, D.: Depth-based hand pose estimation: methods, data, and challenges. *Int. J. Comput. Vision* **126**(11), 1180–1198 (2018)
 14. Tenenbaum, J.B., Silva, V.D., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. *Science* **290**(5500), 2319–2323 (2000)
 15. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3D convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4489–4497 (2015)
 16. Wu, J., Konrad, J., Ishwar, P.: The value of multiple viewpoints in gesture-based user authentication. In: Proceedings of the IEEE Conference on CVPR Workshops, pp. 90–97 (2014)
 17. Xu, C., Tao, D., Xu, C.: A survey on multi-view learning. *CoRR* abs/1304.5634 (2013)
 18. Yan, P., Khan, S.M., Shah, M.: Learning 4D action feature models for arbitrary view action recognition. In: 2008 IEEE Conference on CVPR, pp. 1–7
 19. Yang, S., Do, J., Jang, H., Jung, J., Bien, Z.: Advanced soft remote control system in human-friendliness. In: Symposium on Advanced Intelligent Systems (SCIS and ISIS), pp. 218–222 (2006). *International Journal of Soft Computing and Intelligent Systems*
 20. Makihara, Y., Mansur, A., Muramatsu, D., Uddin, M.Z., Yagi, Y.: Multi-view discriminant analysis with tensor representation and its application to cross-view gait recognition. In: The 11th IEEE International Conference and Workshops on FG, vol. 1, pp. 1–8 (2015)
 21. Zabulis, X., Baltzakis, H., Argyros, A.: Vision-based hand gesture recognition for human computer interaction. In: The Universal Access Handbook, Lawrence Erlbaum Associates, Inc. (LEA), pp. 34.1–34.30 (2009)
 22. Zou, Z., Premaratne, P., Monaragala, R., Bandara, N., Premaratne, M.: Dynamic hand gesture recognition system using moment invariants. In: Proceedings of The 5th International Conference on Information and Automation for Sustainability, pp. 108–113 (2010)



Human Action Recognition Based on Dual Correlation Network

Fei Han^{1,2}, Dejun Zhang^{1,2(✉)}, Yiqi Wu¹, Zirui Qiu², Longyong Wu¹,
and Weilun Huang¹

¹ China University of Geosciences, Wuhan 430074, China
zhangdejun@cug.edu.cn

² Sichuan Agricultural University, Yaan 625014, China
<https://github.com/djzgroup/DualCorrelationNetwork>

Abstract. Learning to capture long-range relations is essential for video recognition. Modern deep learning systems capture spatio-temporal correlation via 3D CNNs. However, it is highly inefficient. In this work, we propose a Dual Correlation Network (DCN) to elaborate the relationship between channels of 3D CNN along with time series. DCN is designed with a self-attention network in the way of a dual path. The channel correlation path selectively emphasizes interdependent channel maps by integrating associated features among all channel maps. The temporal connection path selectively picks temporal relation by integrating associated features among temporal maps. To address action recognition task, we insert the DCN block to MFNet [3]. Experimental results on the Kinetics, UCF-101 and HMDB-51 demonstrate that our network achieves superior performance to the existing state-of-the-art methods on these three datasets.

Keywords: Action recognition · Correlation · Self-attention

1 Introduction

Human action recognition aims to automatically classify the action in a video, and it is a fundamental topic in computer vision with many societal applications such as video surveillance, video retrieval, robot perception, and smart environment/city. Activity videos are spatio-temporal data which are composed of image frames with a specific width/height (XY) concatenated along time axis (T). There are three kinds of classic architecture for action recognition: two-stream CNN [11], 3D CNNs [14–16], and 2D CNNs with temporal models on top such as LSTM [18, 25] and self-attention [22]. Two-stream CNNs capture appearance and motion information with diverse streams, which turn out to be effective for video classification. Yet, it is time-consuming to train two networks and calculate optical flow in advance. 2D CNNs with temporal models usually focus on capturing coarser and long-term temporal structure, but lack capacity of representing finer temporal relation in a confined spatiotemporal region. To

overcome these limitations, 3D CNNs employ 3D convolution and 3D pooling operations to directly learn spatiotemporal features from stacked RGB volumes.

Recognition from videos requires capturing both spatial and temporal information, desirably using learned convolutional kernels. Numerous works have tried to exploit spatio-temporal representations by using 3D CNN, P3D, (2+1)D [9, 15, 16] and so on. However, the progress of these architecture design and representation trend to learn local correlation along input channels which disregarding the hidden information between channel and temporal, The main reason is 3D CNN inherent complexity calculation process and high dimension. Each channel of the feature map represents different semantic information extracted by neural networks and they are extremely relevant, besides video clips contain different periods during action occur, so that the contribution of the video frame to the action recognition is also different.

This work is guided by whether 3D convolutional neural network (CNN) can learn the correlation between time and space very well? Existing methods for action recognition can be summarized into three ways, including hard attention [25], soft attention [25, 26] and self-attention [18, 22]. Considering computational effectiveness and capacity to learn characteristics, we introduced a Dual Correlation Network (DCN) by using self-attention to capture global correlations in the video domain. Compared with model spatio-temporal correlation with two stream inputs [11] or jointly and implicitly with a 3D convolution [15], our DCN aims to simultaneously consider inter-channel correlation information between temporal and channel features, besides, we used an explicit way with a dual-branch unit to represent different levels of concept and information. As showed in Fig. 1, the proposed DCN block is composed of channel correlation path and temporal correlation path. In order to address action recognition task, the block is inserted into MFNet [3], which decompose the 3D CNN with multi fiber network to ease the amount of calculation of 3D kernels. The DCN block equipped lightweight 3D is capable of learning channel-wise dependencies which gives them the opportunity to learn better representations of videos. The corresponding code is available from our community site¹. We evaluated our method on the Kinetics-400, UCF-101 and HMDB-51 datasets. The experiments on action classification task demonstrate the efficacy contributed by DCN. In summary, our contributions are summarized as following:

- (1) A temporal attention module and a channel attention module are proposed to learn the temporal interdependencies of features and model channel interdependencies, respectively.
- (2) A novel correlation block is proposed, by embedding them between 3D blocks, compared to 3D convolution, it provides a global correlation of temporal and channel in video.
- (3) We investigate the effect of our proposed DCN block with three different video datasets, proving its superior performance through comparison with the state-of-the-art on various public benchmarks.

¹ <https://github.com/djzgroup/DualCorrelationNetwork>.

In the rest of the paper, we introduce related work in Sect. 2, and detail about the proposed correlation network in Sect. 3. Experimental setups and experimental results are located in Sect. 4. And the conclusion of this paper and future work in Sect. 5.

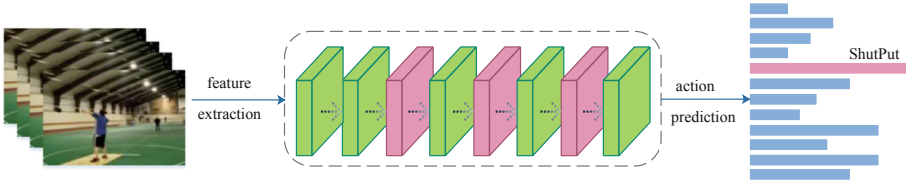


Fig. 1. DCN-MFNet. Our DCN block is inserted into the MFNet. The whole network takes video clips as input, then passed to the feature extractor to extract features, where green represents convolution layer, and pink represents the DCN module. The DCN block operates on the different levels of feature maps in the network. The output of the network is a video-level prediction and different length indicates different labeling values. (Color figure online)

2 Related Work

Learning video representations for human activity recognition have been successful. CNN methods allow end-to-end learning of video features and representations optimized for the training data, performing superior to traditional work [19] for video understanding. In this section, we will briefly review related works involving video action recognition networks and attention models.

2.1 Video Action Recognition Networks

Remarkable progress in action recognition is largely driven by the success of 2D CNNs in image recognition. The original Two-Stream Network [11] take a single RGB frame and a small number of optical flow frames as input to capture both motion and appearance information in videos. Distinct from the image, video possesses temporal structure and motion information, which are important in order to video analysis. This motivates researchers to model them more effectively, such as 3D CNNs [9, 15, 16], Temporal Segment Network (TSN) [21], dynamic image networks [1], and Non-Local Network [22]. Despite the enormous amount of effort on modeling motion via temporal convolution, 3D CNNs can still achieve higher accuracy when fused with optical flow [9, 15, 16], which is unfortunately expensive to compute. Recently, 3D convolutions can be decomposed into a P3D [9] and R(2+1)D [16] or S3D [5]. 3D group convolution was also applied to video classification in ResNeXt [5] and MFNet [3]. Our dual correlation network goes beyond two-stream networks and 3D convolution, and we proposed a new operator that can better learn the temporal dynamics of video sequences and channels of correlation.

2.2 Attention Modules

Attention modules can model long-range dependencies and have been widely applied in many tasks [8, 10, 13, 18]. In particular, [18] is the first to propose the self-attention mechanism to draw global dependencies of input and applies it in machine translation. Meanwhile, attention modules are increasingly applied in image vision field. Zhang et al. [27] introduce a self-attention mechanism to learn a better image generator. The work [22], which is related to self-attention module, mainly exploring the effectiveness of non-local operation in spatio-temporal dimension for videos and images. Different from previous work, we used self-attention mechanism and extended by a two paths of attention modules to capture rich contextual relationships for better feature representations with intra-class compactness. Comprehensive empirical results verify the effectiveness of our proposed method.

3 The Proposed Method

In this section, we describe our method for learning global temporal feature and channels features. We first review the backbone architecture (MFNet [3]). Then we present the dual correlation network in detail. Finally, we consider how to build the dual correlation network to leverage the matching information by incorporating the correlation operator into the backbone. We insert our dual correlation network into different levels of feature maps in the network, Table 1 provides the details of MFNet used in this paper.

3.1 MFNet Backbone

The MFNet [3] was recently introduced and shown to yield state-of-the-art action recognition results on several video datasets. The main idea of MFNet is that the current GFLOPs for 3D CNN networks (such as I3D and R(2+1)D networks) is too high. Commonly used 2D convolutional networks such as resnet-152 or vgg-16 networks are probably 10+ GFLOPs and the two 3D convolutional networks just mentioned have reached 100+ GFLOPs. Therefore, the goal of this work is tantamount to greatly improve the efficiency of 3D CNN model while maintaining the effect of these model.

3.2 Dual Correlation Network

As shown in Fig. 2, a DCN block is a basic computational unit operating on an input volume $A \in \mathbb{R}^{C \times T \times H \times W}$ and an output volume of $D \in \mathbb{R}^{C \times T \times H \times W}$ where H, W, T, C are the height, width, temporal depth and number of channels of the feature maps. The DCN block considers inter-channel correlation information between temporal and channel features and explicit way with a dual path which represents a different levels of concept and information. Specifically, the DCN learns spatiotemporal features from volume input with a two-branch

Table 1. DCN-MFNet. The input size for the network is $16 \times 224 \times 224$, the stride is denoted as (temporal stride, height stride, width stride)

Layer	Repeat	Channel	DCN-MFNet	
			Output size	Stride
Input		3	$16 \times 224 \times 224$	
Conv1	1	16	$16 \times 112 \times 112$	(1,2,2)
Maxpool			$16 \times 56 \times 56$	(1,2,2)
Conv2	1	96	$8 \times 56 \times 56$	(2,1,1)
	2			(1,1,1)
DCN	1	96	$8 \times 56 \times 56$	
Conv3	1	192	$8 \times 28 \times 28$	(1,2,2)
	3			(1,1,1)
DCN	1	192	$8 \times 28 \times 28$	
Conv4	1	384	$8 \times 14 \times 14$	(1,2,2)
	5			(1,1,1)
DCN	1	384	$8 \times 14 \times 14$	
Conv5	1	768	$8 \times 7 \times 7$	(1,2,2)
	2			(1,1,1)
AvgPooling	1		$1 \times 1 \times 1$	
FC	1		101	

architecture: (1) channels correlation path (ccp) for interdependent channel maps learning, and (2) temporal correlation path (tcp) for global temporal dependencies features learning.

Channels Correlation Path. In channels correlation path, each channel maps are feature-specific responses, and different semantics are linked to each other. By exploiting the interdependencies among channel maps, it emphasizes interdependent feature maps and improves the feature representation of explicit semantics. Therefore, a channel correlation module is designed to explicitly model interdependencies among channels. The structure of the channel correlation module is illustrated in Fig. 2. First, we reshape A to $A \in \mathbb{R}^{C, THW}$. Then perform a multiplication between A and the transpose of A . Finally, we apply a softmax layer to obtain the channel correlation $X \in \mathbb{R}^{C \times C}$. We directly calculate the channel correlation map from the feature map. The channel correlation map equation is expressed as follows:

$$X_{j,i} = \frac{\exp(A_i, A_j)}{\sum_{i=1}^c \exp(A_i, A_j)} \quad (1)$$

Where $X^{j,i}$ measures the i^{th} channels impact on the j^{th} channel. In addition, we perform a matrix multiplication between the transpose of X and A and reshape their result to $\mathbb{R}^{C \times T \times H \times W}$. When we multiply the result by a scale

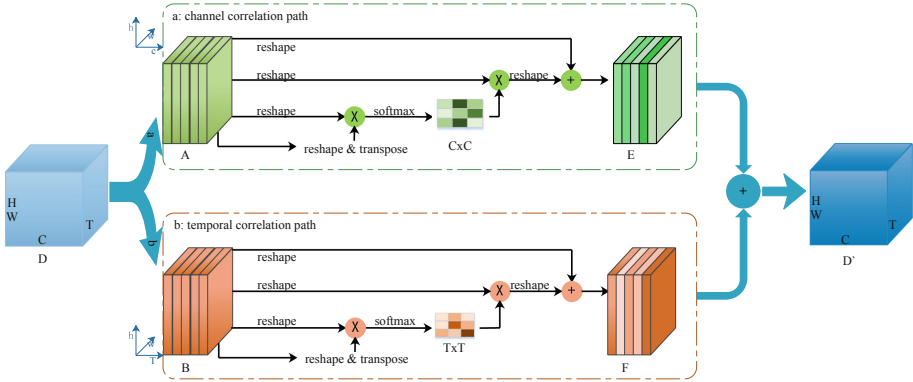


Fig. 2. The details of the channel correlation module (a) and temporal correlation module (b).

parameter α and perform an element-wise sum operation with A to obtain the final output $E \in \mathbb{R}^{C \times T \times H \times W}$ as follows:

$$E = \alpha \sum_{i=1}^c (X_{j,i} A_i) + A \tag{2}$$

Temporal Correlation Path. In temporal correlation path, 3D CNN can learn local spatio-temporal features, but lack of global information, thus we design a temporal correlation path to model the relationships of video frames. The structure of temporal correlation module is illustrated in Fig. 2. As the above method, we reshape B to $B \in \mathbb{R}^{T, CHW}$ and then perform a matrix multiplication between B and the transpose of B. finally we apply a softmax layer to obtain the channel correlation $T \in \mathbb{R}^{T, T}$. We directly calculate the temporal correlation map $T \in \mathbb{R}^{T, T}$ from the feature map $B \in \mathbb{R}^{C \times T \times H \times W}$, the temporal correlation map equation is expressed as follows:

$$T_{j,i} = \frac{\exp(B_i, B_j)}{\sum_{i=1}^c \exp(B_i, B_j)} \tag{3}$$

Where T^{ji} measures the i^{th} position impact on the j^{th} position. In addition, we perform a matrix multiplication between the transpose of X and B and reshape their result to $\mathbb{R}^{C \times T \times H \times W}$. when we multiply the result by a scale parameter β and perform an element-wise sum operation with $B \in \mathbb{R}^{C \times T \times H \times W}$ to obtain the final output as follows:

$$E = \beta \sum_{i=1}^T (T_{j,i} B_i) + B \tag{4}$$

Fusion. In order to take full advantage of long-range contextual information, we aggregate the feature from channels correlation path and temporal correlation path. We perform an element-wise sum to accomplish feature fusion and obtain:

$$D = E + F \quad (5)$$

Where D is correlation volume from A , which learned a significant feature of channels and temporal. The DCN can model the relationship between channels of 3D CNN along with time series, where the channel correlation path selectively emphasizes interdependent channel maps by integrating associated features among all channel maps. The temporal connection path selectively picks temporal relation by integrating associated features among temporal maps. The DCN operates on the video volume and output a video-level prediction. We insert this block into MFNet after conv2-4, thus we can capture global temporal feature and channels feature.

4 Experiments

In Sect. 4.1, we firstly present the action recognition datasets and the evaluation settings. In Sect. 4.2, we study different aspects of our proposed DCN on the Kinetics dataset and compare it with the state-of-the-art methods. In Sect. 4.3, we transfer the learned spatiotemporal representations in DCN to the datasets of UCF-101 and HMDB-51. In Sect. 4.4, we visualize the features we learned from our models.

4.1 Datasets

We evaluate our proposed method on three challenging video datasets with human actions, namely HMDB-51 [7], UCF-101 [12], and Kinetics [6]. Table 2 displays the details of the datasets. For all of these datasets, we utilise the standard training/testing splits and protocols provided as the original evaluation scheme. For HMDB-51 and UCF-101, we report the average accuracy over three splits and for Kinetics, we report the performance on the validation and test set.

Table 2. Details of the datasets used for evaluation. The Clips shows the total number of short video clips extracted from the videos available in the dataset

Datasets	Clips	Videos	Classes
HMDB-51	6,766	3,312	51
UCF-101	12,320	25,00	101
Kinetics	306,245	306,245	400

4.2 Results on the Kinetics Dataset

The Kinetics dataset is the largest well-labeled action recognition dataset. Its current version contains 400 action classes and each category has at least 400 videos. In total, there are around 240,000 training videos, 20,000 validation videos, and 40,000 testing videos. The evaluation metric on the Kinetics dataset is the average of top-1 and top-5 error. The experiment of DCN on this dataset with only RGB input under the setting of training from scratch.

In our experiment, the DCN model is trained on Kinetics with an initial learning rate 0.1 which decay step-wisely with a factor 0.1. The weight decay is set to 0.0001 and we use SGD as the optimizer with a batch size of 1,024. Table 3 summarizes the results of our models and other competing methods on the kinetics 400 datasets, and we can see that our model can get improved compared with many state-of-art models. We are first compared with three baseline methods: (1) CNN+LSTM [17], (2) two Stream [11], and (3) 3D CNNs [2, 3, 16, 24]. We propose DCN significantly outperform these baselines by around 10%. Besides, I3D is also equipped with long-term modeling by stacking 64 frames. Then we compare with the recent state-of-the-art methods, namely MFNet, outperform by 0.8%. From the above observation improved the effectiveness of DCN.

Table 3. Accuracy (%) performance comparison of DCN model with state of-the-art methods on kinetics.

Method	Top1	Top5
Two-stream [11]	63.2%	–
ConvNet+LSTM [17]	63.3%	–
S3D [24]	69.4%	89.1%
I3D-RGB [2]	71.1%	89.3%
R(2+1)D-RGB [16]	72.0%	90.0%
MFNet [3]	72.8%	90.4%
Ours	73.6%	91.0%

4.3 Results on the UCF101 and HMDB51 Dataset

UCF-101 and HMDB-51 are another two popular action recognition datasets, whose sizes are relatively small and the performance on them is very high. The UCF-101 has 101 action classes and 13,320 video clips. We follow the official evaluation scheme and report average accuracy over three training/testing splits. The HMDB-51 dataset is a set of realistic videos from various sources, including movies and web videos. This dataset has 6,766 videos from 51 action categories. Our experiment follows the original evaluation scheme using three training/testing splits and reports the average accuracy. As these two datasets are

relatively minute, we cannot train DCN from scratch and thereby transfer the video representations learned from the Kinetics dataset to them by fine-tuning.

We follow experimental settings in [4, 15, 24] and report the averaged three-fold cross validation accuracy. For training the model on UCF-101 and HMDB-51, we use an initial learning rate 0.005 and decrease it for three times with a factor 0.1. The weight decay is placed at 0.0001 and the momentum is set to 0.9 during the SGD optimization.

First, compared with the ImageNet pre-trained model, Kinetics pre-train can significantly improve the performance on small datasets. Then, we demonstrate that fine-tuning DCN outperforms many competitive baselines. The models in the top two of the table are RGB with optical flow baselines based on two stream networks, including Two-Stream and TSN. We propose DCN significantly outperform these baselines by around 10%. The remaining models in the table are robust RGB-only baselines based on 3D CNNs. From Table 4, We can see that our model obtain a comparable performance to the best performer of RGB-3D.

Table 4. Accuracy (%) performance comparison of DCN model with state-of-the-art methods over all three splits of UCF101 and HMDB51.

Methods	+OF	UCF-101	HMDB-51
Two-Stream [11]	+	88.0%	59.4%
TSN [21]	+	94.2%	69.4%
Resnet-50 [4]		82.3%	48.9%
Resnet-152 [4]		83.4%	46.7%
CoviAR [23]		90.4%	59.1%
C3D [15]		82.3%	51.6%
Res3D [24]		85.8%	54.9%
ARTNet [20]		94.3%	70.9%
I3D-RGB [2]		95.6%	74.8%
R(2+1)D-RGB [16]		96.8%	74.5%
MFNet [3]		96.0%	74.6%
Ours		97.9%	74.8%

4.4 Visualization

In order to better understand the features that channel and temporal features our network learned. We provided some action feature visualize. As shown in Fig. 3, we show three examples from UCF-101 dataset. The first column is Original image, 2–4 is a heat map, and the 5–7 shows the focus map of the DCN. Besides, we predict top3 prediction probabilities to demonstrate the ability of DCN to forecast. Those visualize results in the 2–4 column show that DCN by learning

channel correlation can focus on selectively emphasizes interdependent channel maps. Besides from 5–7 columns represent the focus map of different frames, results shows that the video frames combined different information from different frames, effectively combine them to help with the final action recognition.

Label: HeadMassage

Prediction: top1 HeadMassage prob:0.99219, top2 BlowDryHair prob:0.00373 top3 HairCut prob: 0.00276



Label: YOLO

Prediction: top1 ApplyEyeMakeup prob:0.98219, top2 ApplyLipStick prob:0.01161 top3 HairCut prob: 0.00188



Label: ShutPut

Prediction: top1 ShutPut prob:0.98237, top2 ThrowDiscus prob:0.00508 top3 HammerThrow prob: 0.00427



Fig. 3. Visualization on UCF101. Our model predictions top probability, The first column is Original image, 2–4 is a heat map, and the 5–7 shows the focus map of the DCN.

5 Conclusion and Future Work

In this work, we propose a novel architecture, coined as DCN, for spatiotemporal feature learning in videos. Construction of DCN is based on a dual correlation path which aims to simultaneously consider inter-channel correlation information between temporal and channel features and explicit way with a dual-branch unit which represents different level of concept and information. As demonstrated on the Kinetics dataset, DCN block is able to yield better performance than the 3D convolution and DCN with a single RGB input even outperforms the C3D with two-stream input. For representation transfer from Kinetics to datasets of UCF-101 and HMDB-51, DCN also achieves superior performance to the original MFNet.

For DCN, augmenting RGB input with optical flow also helps to improve performance. However, the high computational cost of optical flow prohibits its application in real-world systems. In the future, we will plan to further improve the DCN architecture to overcome the performance gap between single-stream and two-stream input.

Acknowledgment. This work was supported in part by the National Natural Science Foundation of China under Grants 61702350 and 61802355.

References

1. Bilen, H., Fernando, B., Gavves, E., Vedaldi, A., Gould, S.: Dynamic image networks for action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3034–3042 (2016)
2. Carreira, J., Zisserman, A.: Quo vadis, action recognition? A new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6299–6308 (2017)
3. Chen, Y., Kalantidis, Y., Li, J., Yan, S., Feng, J.: Multi-fiber networks for video recognition. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11205, pp. 364–380. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01246-5_22
4. Hara, K., Kataoka, H., Satoh, Y.: Learning spatio-temporal features with 3D residual networks for action recognition. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3154–3160 (2017)
5. Hara, K., Kataoka, H., Satoh, Y.: Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and imagenet? In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6546–6555 (2018)
6. Kay, W., et al.: The kinetics human action video dataset. arXiv preprint [arXiv:1705.06950](https://arxiv.org/abs/1705.06950) (2017)
7. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: 2011 International Conference on Computer Vision, pp. 2556–2563. IEEE (2011)
8. Lin, Z., et al.: A structured self-attentive sentence embedding. arXiv preprint [arXiv:1703.03130](https://arxiv.org/abs/1703.03130) (2017)
9. Qiu, Z., Yao, T., Mei, T.: Learning spatio-temporal representation with pseudo-3d residual networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 5533–5541 (2017)
10. Shen, T., Zhou, T., Long, G., Jiang, J., Pan, S., Zhang, C.: DiSAN: directional self-attention network for RNN/CNN-free language understanding. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
11. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Advances in Neural Information Processing Systems, pp. 568–576 (2014)
12. Soomro, K., Zamir, A.R., Shah, M.: UCF101: a dataset of 101 human actions classes from videos in the wild. arXiv preprint [arXiv:1212.0402](https://arxiv.org/abs/1212.0402) (2012)
13. Tang, J., Hong, R., Yan, S., Chua, T.S., Qi, G.J., Jain, R.: Image annotation by KNN-sparse graph-based label propagation over noisily tagged web images. ACM Trans. Intell. Syst. Technol. (TIST) **2**(2), 14 (2011)
14. Tang, J., Jin, L., Li, Z., Gao, S.: RGB-D object recognition via incorporating latent data structure and prior knowledge. IEEE Trans. Multimedia **17**(11), 1899–1908 (2015)
15. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3D convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4489–4497 (2015)

16. Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., Paluri, M.: A closer look at spatiotemporal convolutions for action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6450–6459 (2018)
17. Ullah, A., Ahmad, J., Muhammad, K., Sajjad, M., Baik, S.W.: Action recognition in video sequences using deep bi-directional LSTM with cnn features. *IEEE Access* **6**, 1155–1166 (2017)
18. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, pp. 5998–6008 (2017)
19. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3551–3558 (2013)
20. Wang, L., Li, W., Li, W., Van Gool, L.: Appearance-and-relation networks for video classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1430–1439 (2018)
21. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks: towards good practices for deep action recognition. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 20–36. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_2
22. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7794–7803 (2018)
23. Wu, C.Y., Zaheer, M., Hu, H., Manmatha, R., Smola, A.J., Krähenbühl, P.: Compressed video action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6026–6035 (2018)
24. Xie, S., Sun, C., Huang, J., Tu, Z., Murphy, K.: Rethinking spatiotemporal feature learning: speed-accuracy trade-offs in video classification. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11219, pp. 318–335. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01267-0_19
25. Xu, K., et al.: Show, attend and tell: Neural image caption generation with visual attention. In: International Conference on Machine Learning, pp. 2048–2057 (2015)
26. Zhang, D., Luo, M., He, F.: Reconstructed similarity for faster GANS-based word-translation to mitigate hubness. *Neurocomputing* **362**, 83–93 (2019)
27. Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-attention generative adversarial networks. arXiv preprint [arXiv:1805.08318](https://arxiv.org/abs/1805.08318) (2018)



Feature Engineering Workflow for Activity Recognition from Synchronized Inertial Measurement Units

A. W. Kempa-Liehr¹(✉) , Jonty Oram¹, Andrew Wong², Mark Finch³,
and Thor Besier⁴ 

¹ Department of Engineering Science, The University of Auckland,
Auckland, New Zealand

a.kempa-liehr@auckland.ac.nz

² IMeasureU Ltd., Auckland, New Zealand

³ Vicon Motion Systems Ltd., Oxford, UK

⁴ Auckland Bioengineering Institute, The University of Auckland,
Auckland, New Zealand

t.besier@auckland.ac.nz

Abstract. The ubiquitous availability of wearable sensors is responsible for driving the Internet-of-Things but is also making an impact on sport sciences and precision medicine. While human activity recognition from smartphone data or other types of inertial measurement units (IMU) has evolved to one of the most prominent daily life examples of machine learning, the underlying process of time-series feature engineering still seems to be time-consuming. This lengthy process inhibits the development of IMU-based machine learning applications in sport science and precision medicine. This contribution discusses a feature engineering workflow, which automates the extraction of time-series feature on based on the FRESH algorithm (FeatuRe Extraction based on Scalable Hypothesis tests) to identify statistically significant features from synchronized IMU sensors (IMeasureU Ltd., NZ). The feature engineering workflow has five main steps: time-series engineering, automated time-series feature extraction, optimized feature extraction, fitting of a specialized classifier, and deployment of optimized machine learning pipeline. The workflow is discussed for the case of a user-specific running-walking classification, and the generalization to a multi-user multi-activity classification is demonstrated.

1 Introduction

Human Activity Recognition (HAR) is an active research area within the field of *ubiquitous sensing*, which has applications in medicine (monitoring exercise routines) and sport (monitoring the potential for injuries and enhance athletes performance). For a comprehensive overview on this topic refer to [6]. Typically the design of HAR applications has to overcome the following challenges [1]:

© Springer Nature Singapore Pte Ltd. 2020

M. Cree et al. (Eds.): ACPR 2019 Workshops, CCIS 1180, pp. 223–231, 2020.

https://doi.org/10.1007/978-981-15-3651-9_20

1. Selection of the attributes to be measured.
2. Construction of a portable and unobtrusive data acquisition system.
3. Design of feature extraction and inference methods.
4. Automated adjustment to new users without the need for re-training the system.
5. Implementation in mobile devices meeting energy and processing requirements.
6. Collection of data under realistic conditions.

In this contribution, we are discussing the automated engineering of time-series features (challenge 3) from two synchronized inertial measurement units as provided by IMeasureU’s BlueThunder sensor [8]. Each sensor records acceleration, angular velocity, and magnetic field in three spatial dimensions. Due to the availability of machine learning libraries like `tsfresh` [2] or `hctsa` [5], which automate the extraction of time-series features for time-series classification tasks [4], we are shifting our focus from the engineering of time-series features to the engineering of time-series. For this purpose, we are considering not only the 18 sensor time-series from the two synchronized sensors but also 6 paired time-series, which measure the differences between the axes of different sensors. A further focus of this contribution is the optimization of the feature extraction process for the deployment of the machine learning pipeline (Sect. 2). The workflow is discussed for the case of a user-specific running-walking classification (Sect. 3.1), and the generalization to a multi-user multi-activity classification (Sect. 3.2) is demonstrated. The paper closes with a short discussion (Sect. 4).

2 Automated Feature Engineering Workflow

The automated feature engineering workflow presented in this paper has two foundations: The BlueThunder sensor from IMeasureU Ltd. [8] and the time-series feature extraction library `tsfresh` [2,3].

2.1 Synchronized Inertial Measurement Units

The BlueThunder sensor is a wireless inertial measurement unit (IMU), which combines a 3-axis accelerometer, a 3-axis gyroscope, and a 3-axis compass. Its specification is listed in Table 1 and its dimensions are shown in Fig. 1a. One of the key features of this sensor is the fact that several units can be synchronized. Therefore, not only the measured sensor signals itself, but also paired signals, like, e.g. the difference between the acceleration in the x-direction of two different sensors can be used as an additional signal. One might interpret these computed signals as being recorded by virtual sensors, which of course are basically signal processing algorithms.

In order to demonstrate the applicability of the presented feature engineering workflow, we are going to discuss two different activity recognition experiments. The first experiment is concerned with the discrimination of running vs walking

Table 1. Specification of IMeasureU BlueThunder sensor [8].

Features	
Accelerometer range	± 16 g
Accelerometer resolution	16 bit
Gyroscope range	$\pm 2000^\circ/\text{s}$
Gyroscope resolution	16 bit
Compass range	$\pm 1200 \mu\text{T}$
Compass resolution	13 bit
Data logging	500 Hz
Weight	12 g

for a specific person (Sect. 3.1), the second with generalizing the classification of 10 different activities over different persons (Sect. 3.2). The running vs walking classification experiment was designed with a basic setup of two different IMUs being mounted at the left and right ankle. The multi-activity classification task considered 9-different mounting points, which were mounted at the left and right upper arm, the left and right wrist, the left and right ankle, as well as the top of the left and right foot (Fig. 1b).

2.2 Feature Extraction on the Basis of Scalable Hypothesis Testing

At the core of the Python-based machine learning library `tsfresh` [2] is the FRESH algorithm. FRESH is the abbreviation for *FeatuRe Extraction on the basis of Scalable Hypothesis testing* [3]. The general idea of this algorithm is to characterise each time-series by applying a library of curated algorithms, which quantify each time-series with respect to their distribution of values, correlation properties, stationarity, entropy, and nonlinear time-series analysis. Of course, this brute force feature extraction is computationally expensive and has to be followed by a feature selection algorithm in order to prevent overfitting. The feature selection is done by testing the statistical significance of each time-series feature for predicting the target and controlling the false discovery rate [3]. Depending on the particular feature-target combination, the algorithm chooses the type of hypothesis test to be performed and selects the set of statistically significant time-series features while preserving the false discovery rate. The pseudocode of the FRESH algorithm is given in Algorithm 1.

2.3 Feature Engineering Workflow for Activity Recognition

The general approach of the feature engineering workflow for activity recognition has five major steps:

Time-series engineering. Increase the number of time-series by designing *virtual sensors*, which combine the signals from different sensors, compute attributes like derivatives, or do both.

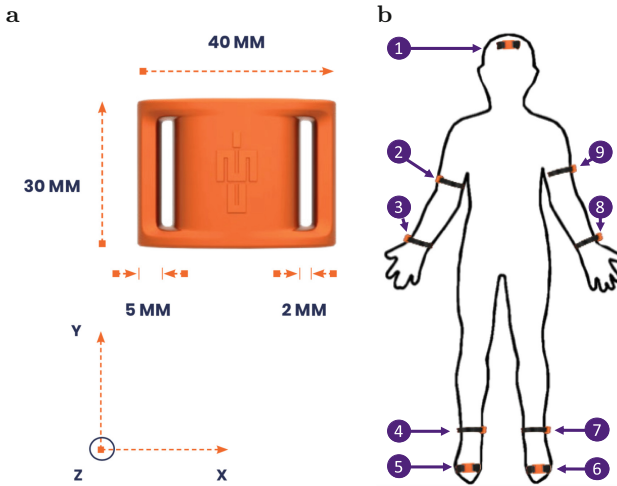


Fig. 1. IMeasureU’s BlueThunder sensor. Panel **a** dimensions of sensor [8, p.2], panel **b** Mounting points of sensors at the front of head (1), left and right upper arm (2, 9), left and right wrist (3, 8), left and right ankle (4, 7), and top of left and right foot (5, 6). For the running-walking classification, sensors were mounted at the left and right ankle (4, 7). For the multi-activity classification, the optimal sensor combination was tip of right foot (5) and right upper arm (2).

Automated time-series feature extraction. Extract a huge variety of different time-series features, which are relevant for predicting the target.

Optimized feature extraction. Identify a subset of features, which optimizes the performance of a cross-validated classifier.

Fitting of specialized classifier. Refit the classifier by using only the subset of features from the previous step.

Deployment of optimized algorithm. Extract only those time series features, which are needed for the specialized classifier.

Note that the deployment step uses the fact that every feature can be mapped to a combination of a specific time-series and a well-defined algorithm. Most likely, not all time-series are relevant and depending on the classifier, only a small set of time-series features is needed. An example of this workflow is documented in the following case-study for classifying running vs walking.

3 Activity Recognition Case Studies

3.1 Running vs Walking

The following case study trains an individualized activity recognition algorithm for discriminating running vs walking on the basis of a 560s long activity sequence, for which the corresponding activities were logged manually:

- 2 synchronized IMUs mounted at left and right ankle (cf. Fig. 1b),
- 560s of mixed running and walking,
- 280000 measurements for each of the 18 sensors (plus 6 paired measurements),
- 140 sections of 4s length (82 walking-sections, 58 running-sections),
- 15605 features in total,
- 4850 statistically significant features (false discovery rate 5%),

The virtual sensor was configured to compute the magnitude of difference between corresponding directions of the acceleration and gyroscope sensors. The time-series features were extracted with `tsfresh` [2]¹, which was available in version 0.10.1 at the time of this case study. A random forest classifier as implemented in `scikit-learn` [7] (version 0.19.0) was used for discriminating running vs walking. The default configuration of the classifier already achieved 100% accuracy under 10-fold cross-validation, such that no hyperparameter tuning was performed. The following 20 time-series features were identified as optimized time-series feature subset as features with the highest feature importances from 100k fitted random forests.

```
[ 'accel_y_diff_agg_linear_trend_ff_agg_max' --chunk_len_5_attr_stderr' ,
  'accel_y_diff_change_quantiles_ff_agg_var' --isabs_True_qh_1.0_ql_0.0' ,
  'accel_y_r_agg_linear_trend_ff_agg_min' --chunk_len_10_attr_stderr' ,
  'accel_y_r_change_quantiles_ff_agg_mean' --isabs_True_qh_1.0_ql_0.0' ,
  'accel_y_r_change_quantiles_ff_agg_var' --isabs_False_qh_1.0_ql_0.2' ,
  'accel_y_r_change_quantiles_ff_agg_var' --isabs_False_qh_1.0_ql_0.4' ,
  'accel_z_diff_change_quantiles_ff_agg_var' --isabs_True_qh_1.0_ql_0.8' ,
  'accel_z_l_agg_linear_trend_ff_agg_min' --chunk_len_10_attr_stderr' ,
  'accel_z_l_change_quantiles_ff_agg_var' --isabs_False_qh_0.6_ql_0.0' ,
  'accel_z_r_minimum' ,
  'gyro_x_r_change_quantiles_ff_agg_var' --isabs_True_qh_0.4_ql_0.2' ,
  'gyro_y_diff_agg_linear_trend_ff_agg_max' --chunk_len_10_attr_stderr' ,
  'gyro_y_diff_agg_linear_trend_ff_agg_max' --chunk_len_50_attr_stderr' ,
  'gyro_y_diff_change_quantiles_ff_agg_var' --isabs_False_qh_1.0_ql_0.4' ,
  'gyro_y_diff_change_quantiles_ff_agg_var' --isabs_True_qh_1.0_ql_0.0' ,
  'gyro_y_l_change_quantiles_ff_agg_var' --isabs_True_qh_0.6_ql_0.4' ,
```

Data: Labelled samples comprising different time-series

Result: Relevant time-series features

for all predefined feature extraction algorithms do

for all time-series do

for all samples do

 Apply feature extraction algorithm to time-series sample and
 compute time-series feature;

end

 Test statistical significance of feature for predicting the label;

end

end

Select significant features while preserving false discovery rate;

Algorithm 1. Pseudocode of Feature extRaction on the basis of Scalable Hypothesis testing (FRESH).

¹ <https://github.com/blue-yonder/tsfresh/tree/v0.10.1>.

```
'gyro_z_l__change_quantiles__f_agg__var' --isabs_False__qh_0.6__ql_0.4',
'gyro_z_r__change_quantiles__f_agg__mean' --isabs_True__qh_0.6__ql_0.4',
'gyro_z_r__change_quantiles__f_agg__mean' --isabs_True__qh_0.8__ql_0.2',
'gyro_z_r__change_quantiles__f_agg__var' --isabs_False__qh_0.6__ql_0.0']
```

These 20 time-series features are computed from 10 different time-series: four from the right ankle (`accel_y_r`, `accel_z_r`, `gyro_x_r`, `gyro_z_r`), three from the left ankle (`accel_z_l`, `gyro_y_l`, `gyro_z_l`), and three magnitude of differences (`accel_y_diff`, `accel_z_diff`, `gyro_y_diff`). Each feature references the generating algorithm using the following scheme [2]: (1) the time-series `kind` the feature is based on, (2) the name of the feature calculator, which has been used to extract the feature, and (3) key-value pairs of parameters configuring the respective feature calculator:

```
[kind] -- [calculator] -- [parameterA]-[valueA]--[parameterB]-[valueB]
```

The features are dominated by two different methods, which quantify the linear trend (`agg_linear_trend`) and the expected change of the signal (`change_quantiles`). A detailed description of the underlying algorithms can be found in the `tsfresh` documentation². The list of features can be converted into a dictionary using the function

```
tsfresh.feature_extraction.settings.from_columns()
```

which can be used for restricting the time-series feature extractor of `tsfresh` to extract just this specific set of time-series features³.

Figure 2a summarizes the feature engineering workflow for the running vs walking case study. The inlay at the bottom right of this figure is also depicted in Fig. 2b. It shows the estimated activity sequence as time-series of probabilities on a hold-out data set, which was recorded by the same person as the training data set but on a different date. For this activity classification, only the 20 time-series features listed above were used. The algorithm's accuracy on the hold-out dataset was 92%.

3.2 Multi-activity Classification Case Study

The following case study involves a more complex feature engineering setup because all nine sensor mounting points, as depicted in Fig. 1, were considered for the feature engineering. The task of this case study was to find a combination of sensors for recognizing the activities

- laying down face down,
- push-ups,
- running,

² https://tsfresh.readthedocs.io/en/v0.10.1/text/list_of_features.html.

³ https://github.com/blue-yonder/tsfresh/blob/master/notebooks/the_fc_parameters-extraction-dictionary.ipynb.

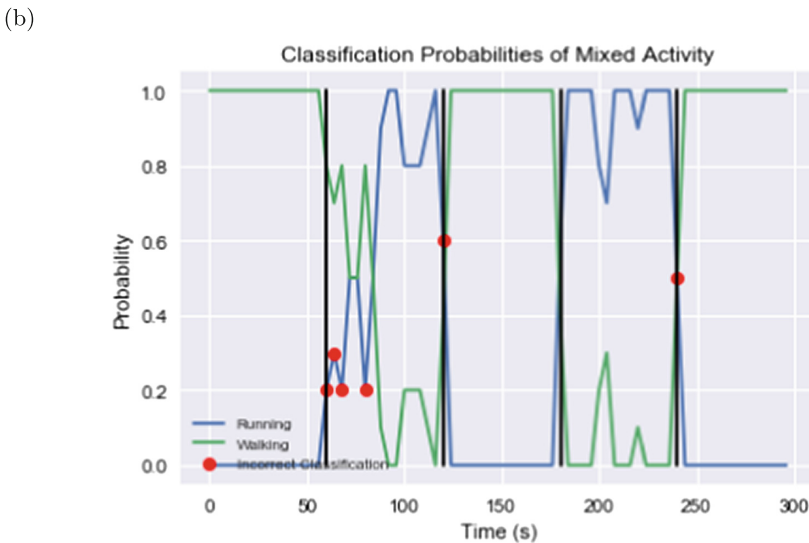
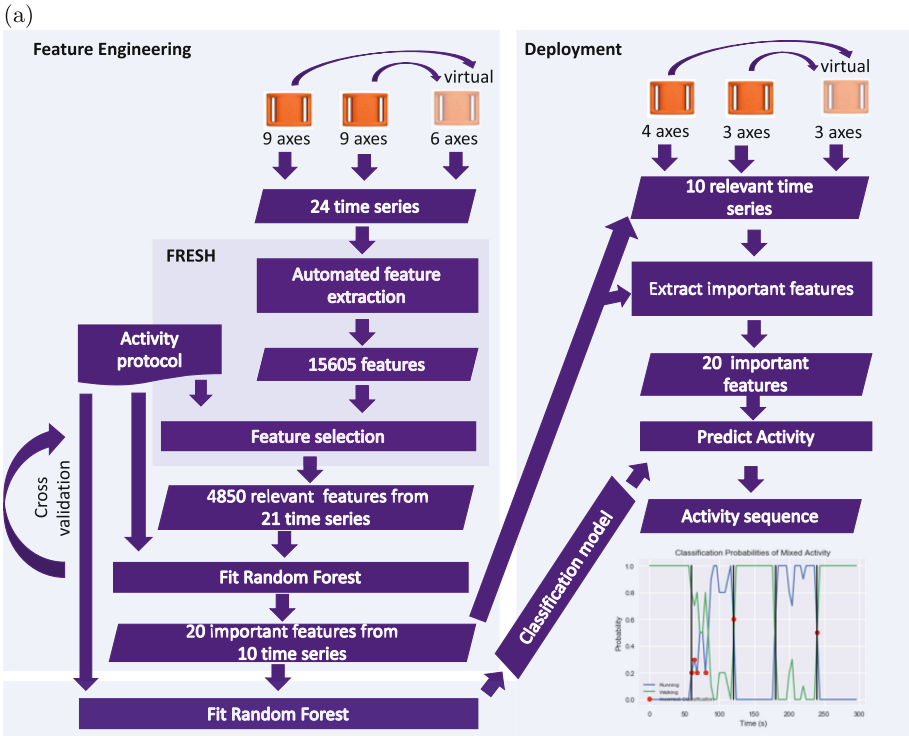


Fig. 2. Feature engineering workflow for activity recognition tasks with details for the running vs walking case study (panel a). Classification of running vs walking for validation data set operating on the 20 time-series features identified during the feature engineering phase of the case study (panel b). Red dots indicate misclassifications. The algorithm has an accuracy of 92%. (Color figure online)

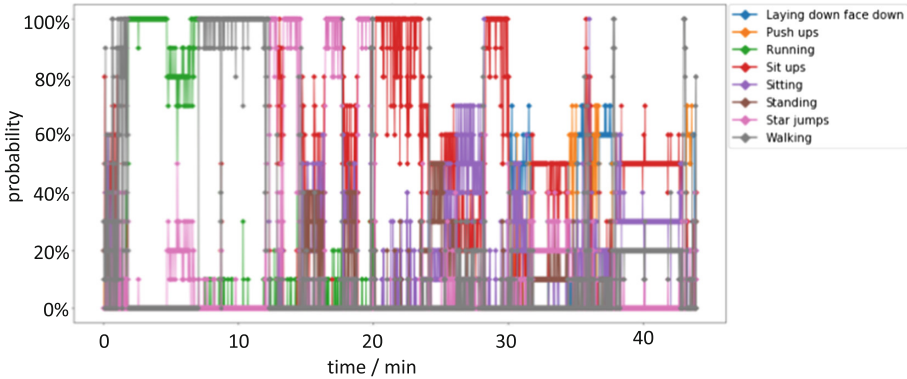


Fig. 3. Evaluation of multi-activity recognition pipeline on the hold-out data set. (Color figure online)

- sit-ups,
- standing,
- star jumps, and
- walking,

while allowing for optimal generalization to other individuals. Therefore, the feature engineering was optimized on the basis of a group 5-fold cross-validation of activities from five different persons (four men, one woman). The mean accuracy for this proband-specific cross-validation was 92.6%.

The optimal sensor mounting points for this task have been identified as the tip of the right foot and the upper right arm (Fig. 1). The evaluation of the resulting activity recognition algorithm on a sixth subject, who had recorded a 45 min long evaluation data set, retrieved a similar performance (Fig. 3) and was computed in less than 20 s.

4 Discussion

The presented workflow for feature engineering of activity recognition task demonstrates a flexible and robust methodology, which is based on the combination of signals from synchronized IMUs and automated time-series feature extraction. Due to the availability of machine learning libraries for automated time-series feature extraction, it can be expected that there will be a general shift of focus in research from the engineering of time-series features to the engineering of time-series. In this work, the engineering of time-series has been modelled as virtual sensors, but in many cases, this process will be similar to the design of signal operators.

Acknowledgement. The authors like to thank Julie Férard and the team at IMeasureU for their support.



References

1. Ahmadi, A., et al.: Toward automatic activity classification and movement assessment during a sports training session. *IEEE Internet Things J.* **2**(1), 23–32 (2015)
2. Christ, M., Braun, N., Neuffer, J., Kempa-Liehr, A.W.: Time series FeatuRe extraction on basis of scalable hypothesis tests (tsfresh - a Python package). *Neurocomputing* **307**, 72–77 (2018). <https://doi.org/10.1016/j.neucom.2018.03.067>
3. Christ, M., Kempa-Liehr, A.W., Feindt, M.: Distributed and parallel time series feature extraction for industrial big data applications. *Learning* (2016). <https://arxiv.org/abs/1610.07717v1>. Asian Conference on Machine Learning (ACML), Workshop on Learning on Big Data (WLBD)
4. Fulcher, B.D.: *Feature-Based Time-Series Analysis*, pp. 87–116. Taylor & Francis, Boca Raton (2018)
5. Fulcher, B.D., Jones, N.S.: hctsa: a computational framework for automated time-series phenotyping using massive feature extraction. *Cell Syst.* **5**(5), 527–531.e3 (2017). <https://doi.org/10.1016/j.cels.2017.10.001>
6. Lara, O.D., Labrador, M.A.: A survey on human activity recognition using wearable sensors. *IEEE Commun. Surv. Tutor.* **15**(3), 1192–1209 (2013)
7. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011)
8. Wong, A., Vallabh, R.: IMeasureU BlueThunder sensor. Sensor Specification 1.5, Vicon IMeasureU Limited, Auckland (2018). <https://imeasureu.com/wp-content/uploads/2018/05/Sensor.Specification.v1.5.pdf>

**Towards an Automatic Data Processing
Chain for Airborne and Spaceborne
Sensors**



Infrared-Image Processing for the DLR FireBIRD Mission

Winfried Halle¹ , Christian Fischer¹, Thomas Terzibaschian¹ , Adina Zell², and Ralf Reulke²

¹ DLR German Aerospace Center, Institute of Optical Sensor Systems, 12489 Berlin, Germany

{winfried.halle,c.fischer,T.Terzibaschian}@dlr.de

² Computer Vision, Humboldt-Universität zu Berlin, Berlin, Germany
{Adina.Zell,Ralf.Reulke}@hu-berlin.de

Abstract. The release of greenhouse gases and aerosols from fires has a large influence on global climate: on average, fires are responsible for up to 30% of anthropogenic CO_2 emissions.

The German Aerospace Center (DLR) is operating the “FireBIRD” constellation, which consists of the two satellite missions TET-1 (Technology Test Platform), and BIROS (Bispectral Infrared Optical System) It is dedicated to scientific investigation of the issues involved as well as to early fire detection from space. The satellite and detector approach is based on proven DLR technology achieved during the BIRD (Bispectral Infrared Detection) Mission, which was launched in 2001 and was primarily used for observation of fires and volcanic activity until 2004. The Payload of TET-1 and BIROS has spectral channels in visible (VIS), near infrared (NIR), mid wave (MIR) and a thermal infrared (TIR) channel. The paper is focused on the processing for TET- and BIROS- Fire- BIRD image data. In the FireBird standard processing chain level 1b and 2a data-products are generated automatically for all users after the data reception on ground. The so called fire-radiative-power (FRP) is one of the most important climate relevant parameters which is estimated by using the bi-spectral method. Two characteristics of the FireBIRD sensors are unique: first, the high radiometric dynamic sensitivity for quantitative evaluation of normal temperatures and high temperature events (HTE) in the same scene. Second, the evaluation of the effective fire area in square meters independent of the recorded number of fire cluster sizes, which is given as the number of pixels per cluster. For certain users, such as firefighters, it is necessary to obtain fire data products (location and temperature) quickly and with minimal delay after detection. In such applications, data processing must take place directly on board the satellite without using a complex processing chain. The paper describes also an alternative fire-detection algorithm which uses artificial neural networks (deep learning) and will compare it with the standard Level-2 FireBIRD processing.

Keywords: Small satellite constellation · Infrared instruments · High temperature events · Bi-spectral method · Artificial-neural-networks

1 Introduction

FireBird is defined as a constellation of two small satellites mainly dedicated to the investigation of high temperature events. The first satellite TET-1 was launched on 12 June 2012. The second satellite BIROS was launched on 22 July 2016.

Both satellites have an identical infrared payload with special design items for detection and measurement of high temperature events in sub-pixel resolution. The payload design is shown in Fig. 1, its main parameters are listed in Table 1.

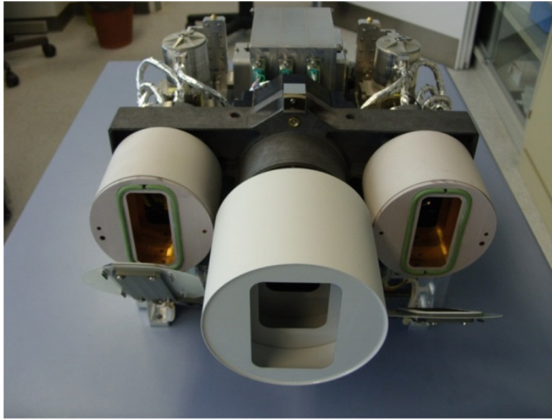


Fig. 1. FireBird camera complex (3 sensor system; left: MWIR, right: LWIR, middle: VIS)

Compared to TET-1 the BIROS Satellite is equipped with a propulsion system to ensure an optimal constellation of the orbits to ensure optimal pointing capabilities. In addition BIROS has a much more powerful data processing system on board, which enables the implementation of a very flexible processing chain.

Due to the limited resources of a small satellite, there are some limitations that can be partially compensated by a flexible operation of the satellite. A well-developed service on demand, especially in the case of BIROS, can significantly improve the satellite's data throughput. All these aspects of using a small satellite to study high temperature events (HTE) are explained below.

2 Remote Sensing and Detection High Temperature Events with Small Satellites

The temporal and spatial distribution of high temperature events (HTE) and the intensity of the events, including their background to the HTE to be taken into

Table 1. FireBird camera parameters.

	CCD line-VIS camera (3 line FPA)	Infrared-cameras
Wave length	1. 460–560 nm 2. 565–725 nm 3. 790–930 nm	MWIR: 3,4–4,2 μm LWIR: 8,5–9,3 μm
Focal length	90,9 mm	46,39 mm
FOV	19,6°	19°
F-Number	3,8	2,0
Detector	CCD-line	CdHgTe staggerd arrays
Detector cooling	Passive, 20° C	Stirling, 80–100 K
Pixel size	7 μm \times 7 μm	30 μm \times 30 μm
Number of pixel	3 \times 5164	2 \times 512 staggered
Quantization	14 bit	14 bit
Ground sampling distance	42,4 m ²	356 m ²
Ground resolution	42,4 m ²	178 m ²
Swath width	211 km ² km	178 km ²
Data rate	Max 44 MBit/s nom 11,2	0,35 MBit/s
Accuracy	100 m on ground	100 m on ground

account for the analysis, can vary considerably. In Oertel [6] different scenarios were examined with regard to the satellite instruments available at that time. Different observation scenarios are triggered not only by the different types of HTEs, but also by the different types of user groups. In this context it is worthwhile to continue the investigation of the advantages and disadvantages of small satellites begun in Lorenz, [5].

For the investigation of such highly dynamic events as bushfires in particular, the re-visit time and the overflight time of the satellite is an important evaluation criterion. Due to the piggy back launch of small satellites the possibility choosing the over-flight-time (LTAN) is limited. The probability to get an ascending node of about 12:00 o'clock or later (A-Train) is very low, but it is needed to have the possibility to detect and evaluate mostly larger fires like the Aqua- and Terra-Satellites from the USA. TET-1 and BIROS have an equator crossing time of 10:30 am and 9:30 am. From these orbits particularly small fires, which develop in the morning, can be detected. The variability is higher with regard to the revisit time. In a standard flight mode (only nadir pointing orientation, without consideration of the off-nadir pointing options), the TET-1 revisit time is approx. 3 or 4 days (maximum at the equator). The second BIROS satellite was able to reduce this time to 1 day.

Additionally the pointing capability of the satellites with a tilting of up to 30° off track can be taken into account. (The 30° limit is due to the atmospheric

damping, not from the satellite). The sub-satellite point moves from day to day about 600 km, but it is possible to obtain an overlap of the image strips of two consecutive days in the order of 100 km by tilting the satellite out of the nadir direction.

With this it is possible to observe a high-temperature event (HTE) in two subsequent days with one satellite. With a second satellite, it would be possible to ensure more than one daily coverage for a given target. This is valid if only night or day time imaging tasks had been taken into account. But in combination of both it is possible with respect to the given orbit constellation to get up to two images of the same target per day. This was demonstrated in different cases.

Another important point is downlink capacity due to limited resources of small satellites, typically equipped with S-Band transmitter. This allows transmitting approximately 100 MB per contact (with the best elevation). Using the maximum number of spectral bands, FireBird (TET or BIROS) generates more than 100 MB, so more than one contact is required to transmit the data completely. Coming back the detection of HTE's itself, most satellite-based methods rely on sensors with a channel in the middle infrared (MIR) atmospheric window. As shown in different other publications (e.g. Lorenz [5]) the MIR spectral channel is the most sensitive to active fires, as it includes the spectral maximum of emitted fire radiation or is close to while the spectral radiance of the background is lowest here.

The FireBIRD detection algorithm is an adaptive so called contextual algorithm which uses the MIR, TIR and NIR and channels which can distinguish after the classification process between:

1. Detection of potentials hot pixels,
2. rejection class of strong sun glints or clouds,
3. rejection class of bright objects,
4. rejection class of warm surfaces and
5. rejection class of cold clouds.

This basis of the sequential algorithm was developed and tested already for the BIRD Satellite Mission. It has been modified and implemented by the FireBIRD mission for operational use. The most important principles were published in [7].

3 BIROS–Satellite Approach

The BIROS satellite is based on a proven approach developed by DLR for the BIRD mission launched in 2001 (Briss, [1]). BIROS satellite bus uses the same technology as the TET satellite, which was successfully launched in July 2012 as the first German “Technology Test Carrier”. TET was initiated and financed by the DLR Space Administration as part of the German On-Orbit Verification (OOV) programm.

At the end of 2013, TET-1 was handed over to the FireBird mission. BIROS was financed by the “BMBF” and was part of the FireBird mission from the beginning.

BIROS and TET-1 use an almost identical multispectral camera system as the main payload. (see Fig. 2). On board BIROS are several other technological experiments designed to contribute to the scientific and technological challenges of the next generation of remote sensing satellites.

High torque reaction wheels and the propulsion system system on BIROS should be emphasized. Particularly in the field of space-based disaster warning systems micro satellites are becoming more and more interesting as highly agile and accurate pointing platforms with the options of swath width extension, in the track stereo imaging, fast multi-target pointing in combination with a high flexibility to command the sensor systems to enable different data acquisition scenarios and finally a fast and flexible distribution of information to the end user on the ground.

BIROS have also a technical on-board experiment by using a hardware VHF modem. Over an ORBCOMM satellite (altitude 800 km) it could be possible to inform directly the ground users via E-Mail about an on-board detected hot-spot with the concerning geo-location e.g for fire-fighters. Here the image classification algorithms will be based on artificial neuronal networks (see paragraph 8).

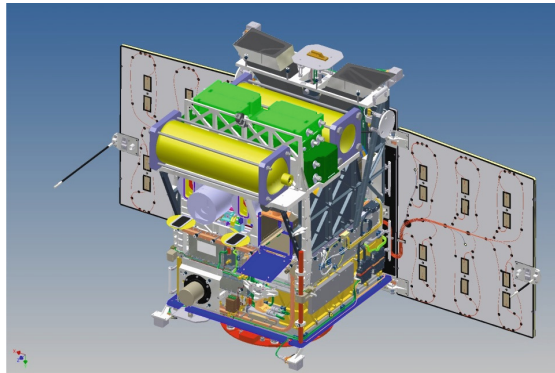


Fig. 2. BIROS satellite with the payload segment

4 Operation and Data Products

The operation of small satellites is differs from that of larger satellites. The FireBIRD-Constellation is a system with a ‘service on demand’ because the limited resources require a very dedicated selection of targets (data-takes). For this operation an individual data ordering process, done by a restricted group of people is necessary. The users can delegate their requests to the order group and in case of conflicts the group decides on the priority of the orders. In addition to the urgency of informing oneself about a disaster situation, the terms conflict

and priority depend above all on technical parameters such as the amount of data to be stored and linked and the number of available ground stations.

Data collection planning is supported by an efficient ordering tool called SPOT, developed by DLR GSOC. The GUI of SPOT is shown in Fig. 3. The use of SPOT also makes it possible to predict future data acquisition and thus to support targeted planning of firing experiments for validation activities and other project-dependent events.

The user can choose from four standard device configurations of the camera to optimally utilize the on-board mass memory (Table 2). During night measurements (Fire Night) the visual bands can be switched off. During the day the GSD of the visual bands can be switched between 40 m and 160 m and for VIS1 it is possible to select which of the visual bands should be transmitted to earth. This is then a decision between the GSD and the area to be monitored on the ground.

Table 2. TET-1 and BIROS standard mode configuration.

Mode	MWiR	MWIR-CAL	LWiR	LWIR-CAL	VISN	VISR	VISG	AOCS	Remark
Fire4 × 4	X	X	X	X	X	X	X	X	GR 160 m
FireNight	X	X	X	X				X	
VIS1 backward	X	X	X	X	X			X	GR 40 m
VIS1 forward	X	X	X	X			X	X	GR 40 m
VIS1 nadir	X	X	X	X		X		X	GR 40 m
VIS3	X	X	X	X	X	X	X	X	GR 40 m
System order	X	X	X	X	X	X	X	X	GR 40 m

When the raw data is received, an operative processing unit generates a standard data format for the raw data. Depending on the operating mode, this L0 level contains up to 5 measurement files, two calibration files for the infrared cameras and a setting file. Based on these raw data files, the L1b standard products are generated. L1b products are radiometrically calibrated data with geographic annotation and associated metadata information. This information can be provided either in an ENVI-compliant data format or in an HDF5 format.

Users are informed about the status of data processing and the products can be downloaded via DLR's EOweb data archiving infrastructure.

5 Radiometric Calibration and Validation

In order to obtain scientifically usable image data, a radiometric calibration of the visible and infrared channels of the sensor must be performed. The application of the corresponding calibration data sets to raw image data is the first step in the image processing pipeline and aims at converting the digital raw image data into units of mean spectral radiation related to the spectral band of each

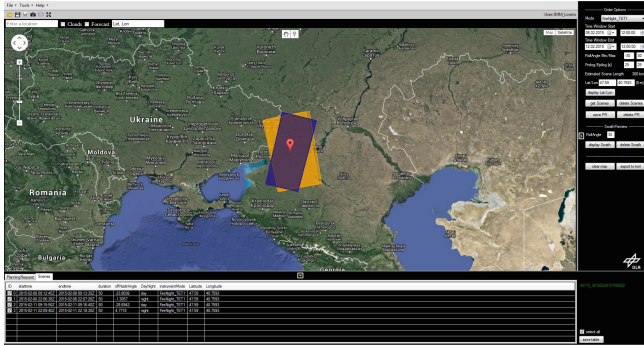


Fig. 3. The GUI of SPOT user tool

channel. For the visible (VIS) and near-infrared (NIR) channels, calibration data sets were obtained from ground-based flat field measurements with well characterized reference sources. These data sets are applied to the incoming VIS/NIR raw data. For the mid-wave infrared (MWIR) and the long-wave-infrared channels (LWIR), calibration data are recorded during flight after data acquisition.

The calibration procedure in the case of FireBird is not a classic two-point procedure, but is based on a correlation of the detector signals with the continuous heating process of the blackbody flap in front of the IR optics (see left Fig. 4 and right Fig. 4).

These efforts concentrate on the development of algorithms for the reconstruction of scene signal sections with very high signal dynamics. For example, very hot temperature events on relatively cold backgrounds in the MWIR and LWIR channels can show nonlinear signal responses, especially at very low signal levels or even information loss due to optical distortion. The described calibration procedure can check the linearity of the signal responses and based on the knowledge of the spatial distribution of the incoming radiation information in the images (so-called Point Spread Functions, PSF), lost information can at least partially be estimated or reconstructed. The result is an increase of the effective dynamic range of the sensor channel. In addition, the signal dynamic range can be increased by operating the system in a special mode with reduced integration time for very hot scenes.

6 Hot Area Technology and the Bi-Spectral Algorithm

The observation of HTE and in particular of wild fires places high demands on the dynamic range which the device has to cope with. An extreme example is the observation of the Bardabunga volcano on the island (see Fig. 5) especially in the MWIR band. In this image, ice and fire stand side by side and the fire fills a series of detector pixels, which requires a very high saturation temperature. On the other hand, the glacier's low IR signal forces the detector into the non-linear operating range as described above. Both extremes can be overcome with

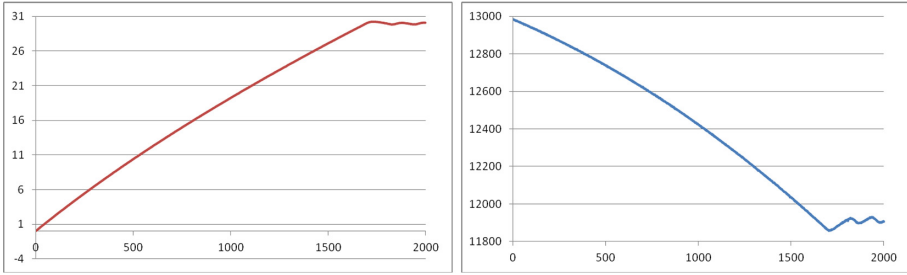


Fig. 4. Left: calibration Black-Body-Flap temperature curve. Right: detector signal (DN) correlated to the flap temperature

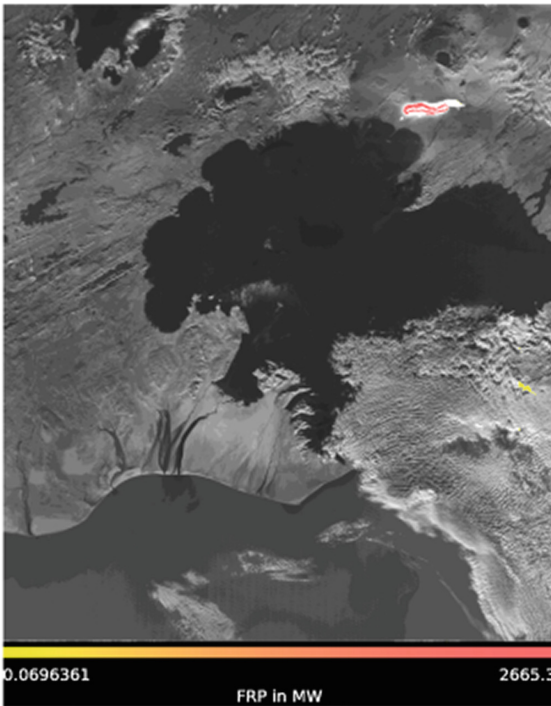


Fig. 5. Fire and Ice-The Bardabunga volcano on Iceland

extremely different integration intervals of the infrared detectors. This is exactly the technology implemented in FireBIRD's IR cameras. The dwell time of the IR cameras is approx. 20 ms, the integration interval for the background temperatures $\sim 20^{\circ}\text{C}$ is set to 6 ms. Controlling the signal levels of the first standard exposure during the readout process in real time makes it possible to initiate a second exposure with a much shorter integration interval in case of saturation. This technology allows an extremely high dynamic range to be covered.

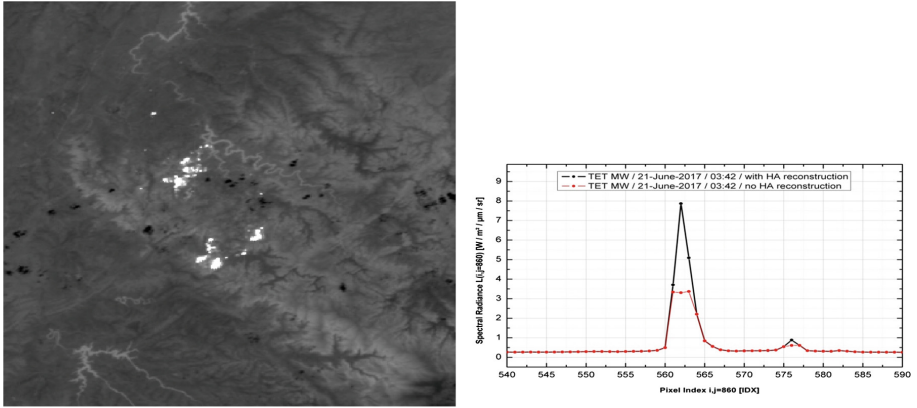


Fig. 6. Left: MWIR Portugal forest fires, 21 June 2017. Right: MWIR spectral radiance of the image line by fire without HA mode (red-line); with HA Mode (black-line) (Color figure online)

The following pictures show an example of this unique on-board function:

In figure Fig. 6 (left) shows the data of the MWIR channel of the Portugal Forest Fire, 21 June 2017. The applied spectral radiance (right) shows the saturated pixels in one of the fire clusters (red line). The black line shows the real signal modulation of the fire cluster after the second data acquisition with the shorter integration time of 500 μs instead of 6ms. After the combination of the two data recordings on the ground, the high dynamic range is also visible in the processed fire clusters. Figure 7 visualizes this function in detail: The left image is based on the processing without the HA processing mode. Here no inner structure of the fire cluster is visible (all pixels are saturated). The right image shows the result in HA processing mode. Here you can distinguish details of the spectral radiation within the fire clusters.

In case of very small a hot spot which covers only a part of an image ground pixel the relating detector signal is a mix of the background temperature and the high temperature resulting in a brightness temperature of may be ~40 °C (for the complete pixel). After the detection of the HTE (see paragraph 2) calculation of the hot spot temperature from this brightness temperature the Dozier method (Dozier, [2]) will be applied.

In the single pixel case, the effective firing temperature T_F and the proportion of fire in the pixel q_F , which refers to the fire section A_F , are determined by solving the mixed equations for the pixel-averaged radiation in two channels:

$$L_j - L_{j,bg}^h = q_F (B_j(T_F^p) - L_{j,bg}^h) \tag{1}$$

where I_j is the atmospherically corrected radiance of a hot pixel in channel j ($j = \text{MIR}$ and TIR), $B_j(T)$ is the black-body radiance in channel j as a function of temperature T , $L_{j,bg}^h$ is the radiance of the non-fire portion of the hot pixel to be estimated from the adjacent background pixels.

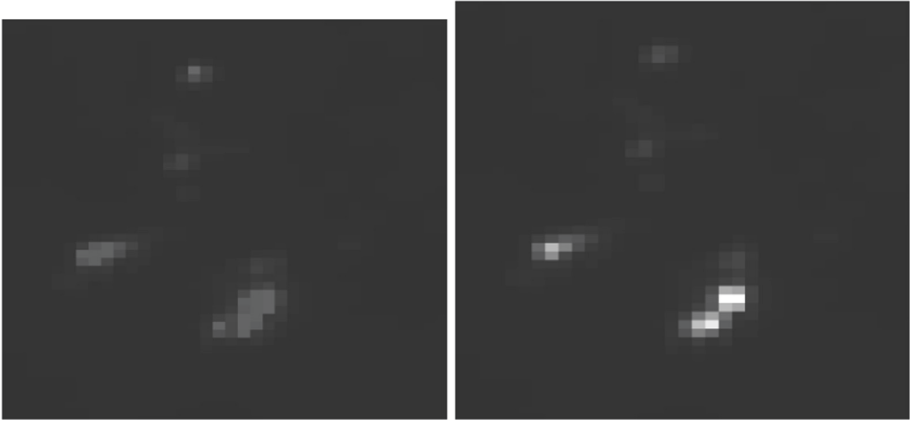


Fig. 7. Left: Fire cluster without HA-Processing. Right: Fire cluster with HA-Processing

Because the MIR radiation intensity of a fire is so intense, even smaller subpixel fires will significantly affect not only the beam signal of the pixel in which the fire is actually located, but also the signal of neighboring pixels. This effect is particularly pronounced in TET-1 and BIROS images, as the double scan causes the pixels to overlap by 50%. For this reason, active fires are usually recognized as clusters of ‘hot’ pixels in MIR imaging, referred to here as ‘hot clusters’.

The area of a hot cluster in an image should not be confused with the area of the causing fire. The Fig. 8 shows an example of the bush fires detected in August 2019 (data-take FBI_BIROS_20190825T020016). The left image shows the spectral radiation of the MIR channel. On the right side the calculated fire radiant power (FRP) is shown. Figure 9 shows on the left side the position of the footprint of the data take over Brazil and on the right side the details (FRP) of some fire clusters of Fig. 8. The ‘Size’ column in the Table 3 from the standard Level 2 data products of the scene in Fig. 8 indicates how many pixels are contained in the cluster described by a row of this table. The ‘Pixel area’ column indicates the area size of the affected pixels in the cluster, but the actual size of the fires is generally smaller: column A (m^2) indicates the effective size of each cluster calculated using the bi-spectral method.

This type of Level 2 data product also helps to locate the bush fire directly and only transmit relevant fire parameters to the local authorities or directly to the fire brigade.

7 FireBIRD Application for Wild Fire Monitoring

Since the IR camera systems on board TET-1 (2012) and BIROS (2017) are active, about 6000 scenes of forest fires or other HTEs, i.e. volcanoes and

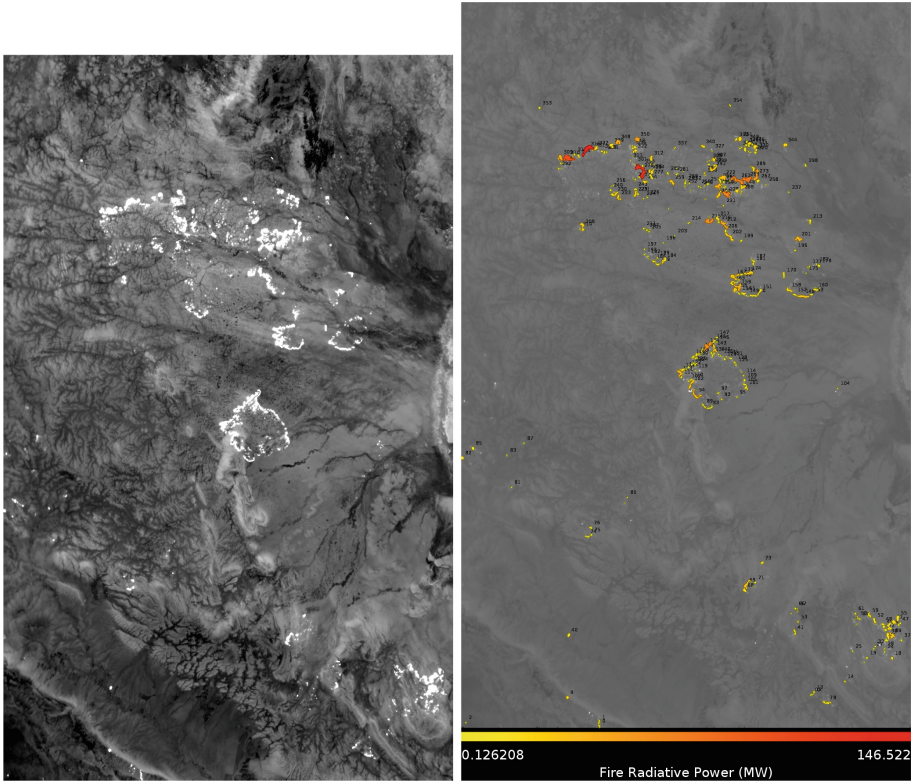


Fig. 8. Left: MIR channel Brazil BIROS 25th August 2019; right: Calculated FRP (Fire Radiative Power)

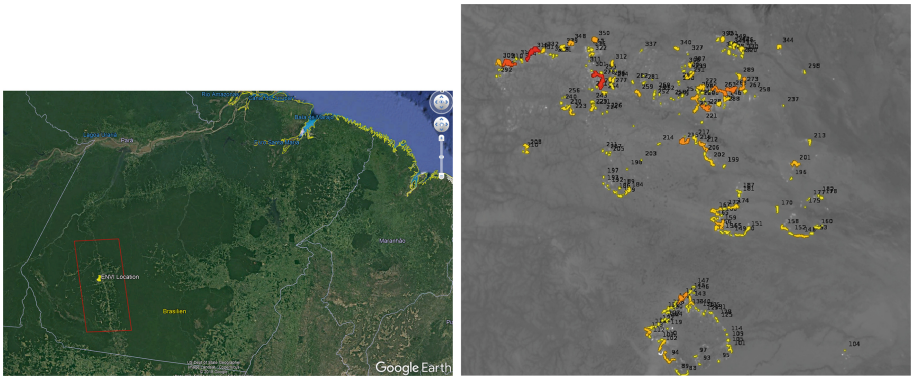


Fig. 9. Left: Location of the footprint of the data-take over Brazil 25th August 2019; right: details (FRP) of some fire-clusters of Fig. 8

Table 3. FRP table for a small field of view of scene FBI_BIROS_20190825T020016.

CustesNO	Size (Pixel)	Pixel area (m ²)	Lat	Long	FRP (MW)	T (K)	A (m ²)
0	15	496860	-18,509	-59,319	5,22	499,3	1480,6
1	21	695604	-18,496	-59,322	8,08	520,5	1941,7
2	5	165620	-18,545	-59,810	0,93	549,3	180
7	44	1457456	-18,336	-58,439	12,01	620,7	1427
8	8	264992	-18,335	-58,424	2,43	560,7	433,8
9	29	960596	-18,421	-59,463	21,71	599,2	2969,5
10	9	298116	-18,314	-58,511	2,49	505,2	672,8
12	4	132496	-18,312	-58,504	0,7	548,5	137,4
13	12	397488	-18,303	-58,500	2,82	513,1	716,7
14	9	298116	-18,250	-58,379	2,55	621,6	300,9
17	3	99372	-18,174	-58,308	0,81	493,6	241,1
18	20	662480	-18,144	-58,188	10,56	563,8	1842,8
19	12	397488	-18,153	-58,296	1,73	666,9	154,5
24	7	231868	-18,123	-58,225	0,84	693,5	63,9
25	6	198744	-18,136	-58,363	1,59	588,9	233
27	7	231868	-18,120	-58,275	1,78	523,5	418,5
28	5	165620	-18,113	-58,228	0,88	673,6	75,2

industrial sites such as power plants, offshore gas and oil platforms, refineries and mines, have been recorded. HTEs occur on all continents and in a wide variety of land cover types, from grasslands in South Africa, eucalyptus forests in Australia, boreal forests in Canada and even volcanoes in Iceland. In Fig. 10 an overview shows the most important placements of the worldwide data recordings with TET.

A very sad example was the devastating bushfires in the USA (Paradise California) in November 2018. TET-1 and BIROS were able to jointly demonstrate the benefits of a constellation of more than one satellite. A time series of both satellites within 5 days shows the change detection capabilities of the FireBIRD system. The Fig. 11 shows the foot-print of the data-takes. In the first days between 10th, 12th and 14th November, the detected fire with the calculated FRP is clearly visible in the Level 2 data products (see Fig. 13).

In the Fig. 13 the combination of 3 data-takes are shown the moving of the fire-fronts. This map projection of the fire data was developed by the DLR-ZKI (DLR-Center for Satellite based Crisis Information) especially for the regional authority and for the fire fighters (Fig. 12).

The Table 4 gives an overview of the time series of the relevant fire parameters of the several days of recorded data-takes. It shows on November 12th the highest fire activation with the total size of the clusters (pixel size = 2241) and the effective size of the fires $A_F = 520645 \text{ m}^2$. On November 14th the fire burns drops ($A_F = 88345 \text{ m}^2$), but the evaluated temperature is even the highest in

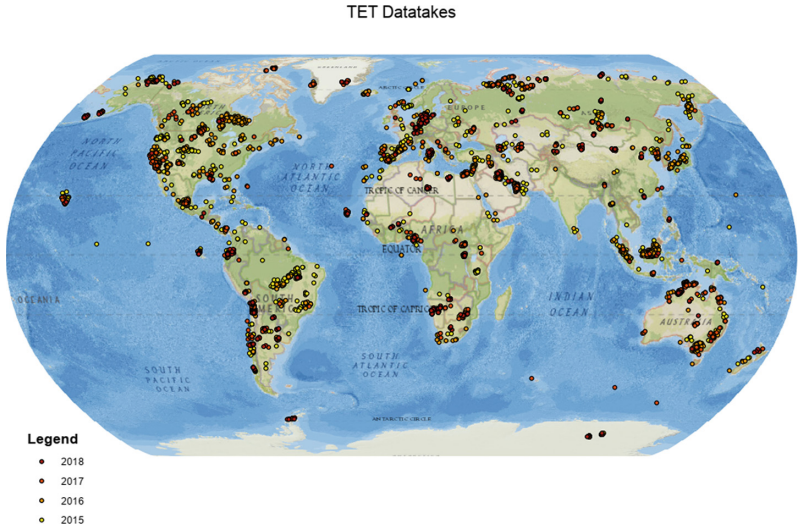


Fig. 10. Overview of the placements of the worldwide data acquisition of the FireBIRD mission until 2018

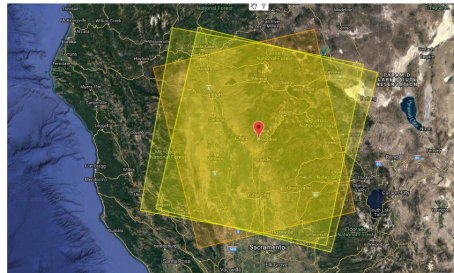


Fig. 11. Overview of the foot-print of the data-takes over Paradise (Nov 2018)

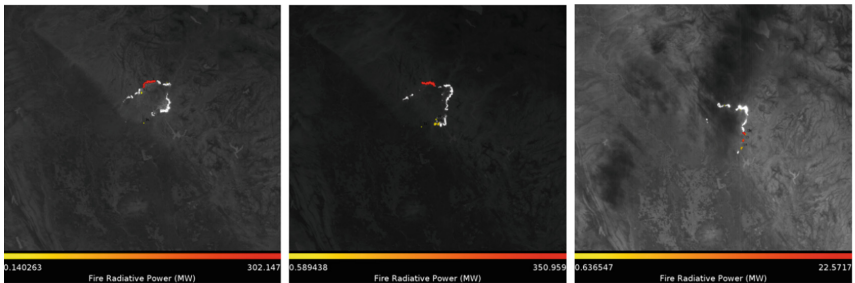


Fig. 12. Left: FRP: 10th Nov/2018. Middle: FRP: 12th Nov/2018. Right: FRP: TET 10th Nov/2018

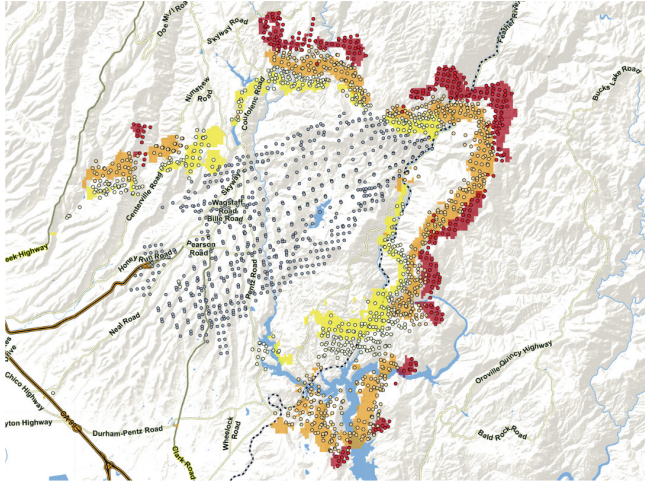


Fig. 13. Fire-Fronts: yellow (10th Nov), orange (12th Nov), red (14th Nov); Source: DLR-ZKI (Color figure online)

the time series of the data acquisition. The local movement of fires between 10th and 14th November is illustrated in Fig. 13 (Source DLR-ZKI).

Table 4. Overview of Level 2 data products for time series of bushfires in Paradise (USA) 2018

Date time/h sensor	10.11 00:19 TET/night	10.11 13:09 TET/night	12.11 13:14 TET/day	14.11 13:19 TET/day	21.11 0:17 BOS night	22.11 18:02 BOS day
FRP/MW	2792	3327	4953	1089	69.7	3.1
T _{fire} /K	714	672	658	752	720	891
A _F /m ²	202750	314956	520645	88345	191	52
Total size cluster/pixel	1511	1738	2241	1173	188	17

8 Preparation of an On-Board FireBIRD Application for Wildfire Monitoring

In the last section, the evaluation of fire-fighting products was demonstrated using some examples. These products were processed with the FireBIRD Level 2 processor in a processing chain at the DLR ground station in Neustrelitz near Berlin after each downlink of the satellite data. The processing time for the data

acquisition can vary up to several minutes depending on the number of recorded fire locations. The reason for this is that for the calculation of the fire radiant power (FRP), co-registration between the MWIR and LWIR channels must be very precise using an adaptive adaptation algorithm.

Especially for firefighters only the detection of fires with the parameters location, size and temperature is of importance. They do not need the FRP, but they need the fire information as quickly as possible. The BIROS satellite has the ability to send this type of information directly to the end user, without using the usual ground stations, via an integrated Orbcocom modem. On the other hand, BIROS has a powerful on-board payload computer for image data processing.

For on-board classification to generate dedicated information for the fire brigade, neural networks are predestined to solve the problem. This has already been demonstrated on the forerunner mission BIRD (see Halle [4]).

Before implementing an artificial neural network on the BIROS payload computer, the algorithms in MATHWORKS were simulated and evaluated offline using different data sets from the FireBIRD archive. The approach of the artificial neural network was carried out as follows: The input image of 12×12 pixels on two channels (MWIR and LWIR) passes through the 16 layers of the neural network, including three convolution layers. It is simply divided into two trained classes, either there is a fire or there is no fire. The output is the probability that the input data belongs to one of these classes.

To detect fire in an image, it is divided into overlapping 12×12 pixel patches. These are classified individually. If a fire on a patch is very likely, then the corresponding coordinates are marked on a mask and the coordinates of the pixel with the highest intensity are stored. At the end of the entire classification process, there is a mask for the entire image on which each pixel on which there is a fire is most likely marked. In addition, a table with the coordinates of possible fire clusters is created (Figs. 14 and 15).

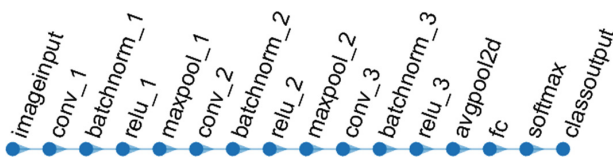


Fig. 14. Topology of the neural network

Table 5 topology of the neural network with 16 layers and 2 trained classes.

The training was carried out on only five different sample pictures. The training images were divided into 98420 12×12 pixel patches. A series of patches from 2018 contained fires. Two of the images were taken in January 2018 in the Niger Delta (TET1 2018/Jan/02, TET1 2018/Jan/04) while there were some fires. Two more show wild fires in California in December 2017 (TET1 2017/Dec/10) and November 2018 (TET2018/Nov/12). The last one was recorded when a volcano erupted on the Galapagos Islands in July 2018 (TET1 2018/July/03). The

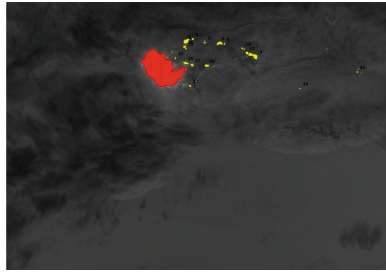


Fig. 15. TET1 2017/Dec/10

ground-truth for the training was based on masks generated by the standard fire-detection-algorithm and then completed by hand (Figs. 16 and 17).

The standard FireBIRD fire-detection algorithm and the neural network were compared based on detected cluster centers and assumed cluster centers, respectively. The image section on which this comparison was performed is part of a scan showing the forest fires in the Amazon rainforest in Bolivia on 24th 2019 (BIROS 2019/Aug/24). 45 common clusters were found by the algorithms and 41 by the neural network (see Table 5).

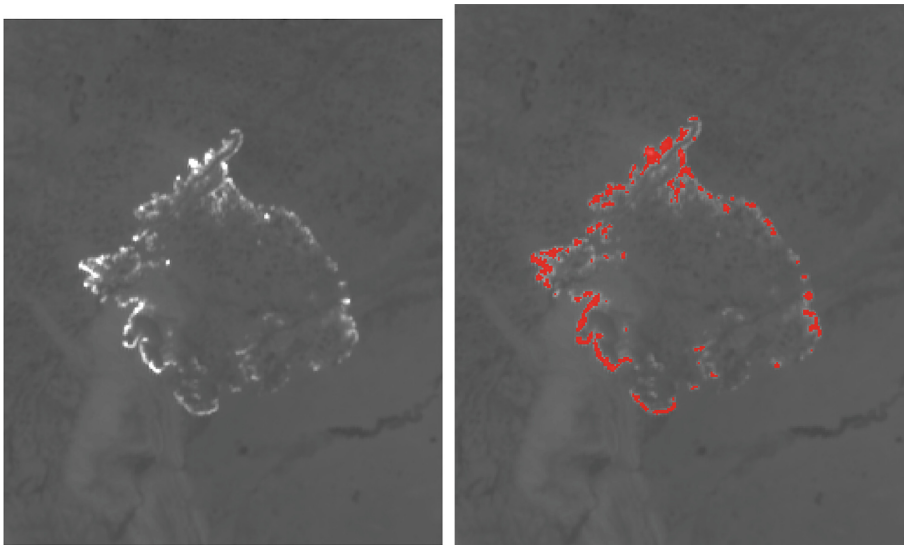


Fig. 16. Left: BIROS Data-Take 2019/Aug/024; right: results from fire-detection-algorithm

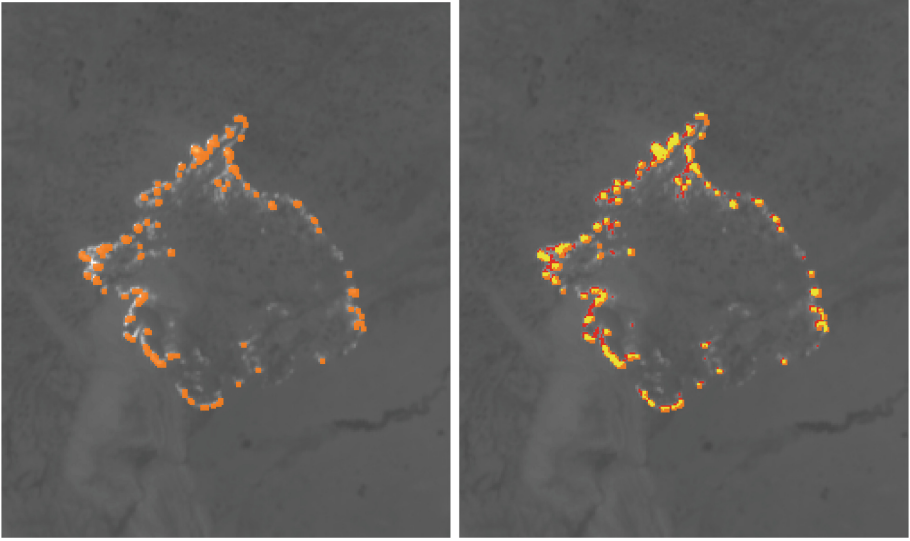


Fig. 17. Left: results of neural network classification; right: combined result; yellow marked pixels were found by both algorithms (Color figure online)

Table 5. Certain results from the cluster centre comparison

Index	Fire-detection-algorithm		Neural net classification	
	mean_x	mean_y	mean_x	mean_y
89	551.5	919.1	552	919
95	627	899.3	629	900
134	571.5	801.5	571	801
144	559.9	781.9	560	783

9 Conclusion

In this paper we have demonstrated the capabilities of the small satellite system FireBIRD to detect high-temperature events with the multispectral infrared sensor system. The sensor system flies on the TET and BIROS satellites of an attitude of approx. 500 km SSOrbit and is capable of detecting and evaluating even small fires with low energy as well as very hot and bright fires by means of an adaptive, so-called hot area-mode when switching the integration time. The benefits of a fire constellation (at least with two satellites) were shown by means of examples of the fire catastrophe in Paradise (USA) in November 2018: Here, data recordings from TET and BIROS show the change of the fire location and the fire parameters (fire size, fire temperature and fire radiation power) during a narrow time series. Another unique feature of FireBIRD's Level 2 processing was demonstrated by a data recording of the giant Amazon rainforest

fire in August 2018: By using the bi-spectral method, the effective cluster fire size (in the sub-pixel range) can be better estimated than using only the number of affected fire pixels of the cluster. For scientific users, FireBIRD standard processing offers Level 2 data products using the bi-spectral method (Dozier). In this case, the fire-radiative-power (FRP) is an important performance value as a climate-related parameter. Firefighters in particular only need information about the location and size of the fire. This information is required as soon as possible after detection and is preferably generated directly on board the satellite. Artificial neural networks are predestined for this application and can be implemented directly in hardware on board the satellites. In this paper the feasibility of the neural network classification of a “fire” class and a “non-fire” class was demonstrated (on the ground) and compared to the results of the bi-spectral method in a FireBIRD data acquisition of the large bushfire in Brazil and Bolivia in August 2019.

References

1. Briess, K., Jahn, H., et al.: Fire recognition potential of the Bi-spectral InfraRed Detection (BIRD) satellite. *Int. J. Remote Sensing* **24**(4), 865–872 (2003)
2. Dozier, J.: A method for satellite identification of surface temperature fields of subpixel resolution. *Remote Sens. Environ.* **11**, 221–229 (1981)
3. Halle, W., Bärwald, W., Raschke C., Terzibaschian T.: The DLR-satellite BIROS in the FireBIRD mission. In: 4S Symposium, Porto Petro, Majorca, Spain (2014)
4. Halle, W.: Ausgewählte Algorithmen der Segmentierung und Klassifikation zur thematischen Bildverarbeitung in der Fernerkundung. DLR-Forschungsbericht (1999)
5. Lorenz, E.: Thermal remote sensing with small satellites: BIRD, TET and next generation BIROS. In: Kuenzer, C., Dech, S. (eds.) *Thermal Infrared Remote Sensing. RDIP*, vol. 17. Springer, Heidelberg (2013). https://doi.org/10.1007/978-94-007-6639-6_8
6. Oertel, D.: ECOFIRE study on scientific assessment of space-borne high temperature event observing. Mission Concepts. ESTEC Contract No. 17690/03/NL/FF, Final Report (2005)
7. Zhukov, B., Lorenz, E., Oertel, D., Wooster, M., Roberts, G.: Experience of detection and quantitative characterization of fires during the experimental small satellite mission BIRD. Final Report to the BIRD Mission, DLR (2005)



Temperature Dependence of Dark Signal for Sentinel-4 Detector

Ralf Reulke¹✉ , Michael P. Skegg², and Rüdiger Hohn²

¹ Institute of Optical Sensor Systems, German Aerospace Center,
12489 Berlin, Germany
ralf.reulke@dlr.de

² Airbus Defence and Space, Robert-Koch-Str. 1, 82024 Taufkirchen, Germany
{[michael.skegg](mailto:michael.skegg@airbus.com),[ruediger.hohn](mailto:ruediger.hohn@airbus.com)}@airbus.com

Abstract. The Sentinel-4 payload is a multi-spectral camera system which is designed to monitor atmospheric conditions over Europe. The German Aerospace Center (DLR) in Berlin, Germany conducted the verification campaign of the Focal Plane Subsystem (FPS) on behalf of Airbus Defense and Space GmbH, Ottobrunn, Germany. In this publication, we will present in detail the temperature dependence of dark signal for the CCD 376 (NIR) from e2v. Dark current is strongly temperature-dependent and is based on the thermal excitation of electrons in the conduction band. During the testing the temperature of the detectors was varied between 215K and 290K. Different models were examined, and corresponding deviations determined. Presenting the dark current by means of an Arrhenius plot, it can be shown, that the activation energy is about half of the band gap. As an important result it could be shown that the temperature dependence can be described by two activation energies.

Keywords: SENTINAL-4 · Dark signal · Temperature dependence · Activation energy

1 Introduction

The Sentinel 4 instrument is an imaging spectrometer, developed by Airbus under ESA contract in the frame of the joint European Union (EU)/ESA COPERNICUS program with the objective of monitoring trace gas concentrations. The German Aerospace Center, DLR Berlin, conducted the verification campaign of the Focal Plane Subsystem (FPS) during the second half of 2016.

The Sentinel-4 Focal Plane Subsystem (FPS) consists of two FPAs, two Front End Electronics (FEEs) and one Front End Support Electronic (FSE). The FPAs house the CCD detectors, the detector-close electronics, as well as internal LEDs for radiometric on-board calibration.

Details of the Sensor and the FPA can be found in the publication from Swindells [13], Hohn [5], Hinger [4] and Hermsen [3]. Information about the

comprehensive test campaign are provided in Skegg [12], Williges [15] and Candéias [1].

As part of this test campaign, the dark signal was extensively investigated as an important performance parameter. Dark signal is varied by changing either the integration time or the operating temperature [7]. When measuring the dark signal with increasing integration times a non-linear behavior was observed for Sentinel-4 Detector by Skegg (see [12] Fig. 6).

This presentation focus on the dark signal variation by temperature change of the NIR detector, because UVVIS detector has a similar behavior. The temperature range varies in this experiment from 215K (operating temperature of the detector) to 292K.

This publication is structured as follows. It begins with an overview of the results of previous work. The following chapter summarizes the known information about the dark current. Then the measurement setup and the procedure will be described. The next chapter provides the results. Finally, conclusions and an outlook are presented.

2 Related Work

A detailed investigation of the dark current for a backside-illuminated CCD was published in the paper from Widenhorn [14]. The temperature range was similar to our experiment (222K to 291K). First the authors assumed that the dark current follow the Arrhenius law, described by the activation energy and a prefactor. The temperature dependencies can be approximated and thus the dark current can be scaled to an arbitrary temperature. In the paper from Popowicz [10] similar results are published.

Then the relation between the prefactor and the apparent activation energy, as described by the Meyer-Neldel rule, was investigated. A more detailed analysis shows that the activation energy for the dark current changes in the temperature range. As a result they stated the relative importance at high temperatures of the diffusion dark current and at low temperatures by the depletion dark current. The diffusion dark current, characterized by the band gap of silicon, is uniform for all pixels. At low temperatures, the depletion dark current, characterized by half the band gap, prevails, but it varies for different pixels.

Dark current spikes are pronounced at low temperatures and can be explained by large concentrations of deep level impurities in those particular pixels. We show that fitting the data with the impurity concentration as the only variable can explain the dark current characteristics of all the pixels on the chip.

3 Dark Current

3.1 Signal Model

An overview about sensor performance consideration can be found in the books from Kopeika [8], Janesick [6] and Janesick [7]. Detector verification for the

Sentinel-4 CCD-Matrix is described in the paper from Williges [15]. Eckardt [2] published a similar investigation for the DESIS [9] detector.

The detector signal model is based on the following approach for the average signal (see Schwarzer [11])

$$\langle s \rangle = \eta_{DV} \cdot \eta_V \cdot \eta_\lambda^{qu} \cdot \tau \cdot A_{pix} \cdot \frac{\lambda}{hc} \cdot E + DS \quad (1)$$

$\langle s \rangle$	Camera output signal [DN]
h	Plank's constant [Js]
c	Speed of light [m/s]
λ	Center wavelength of incident light [m]
$G_S = \eta_{DV} \cdot \eta_V$	Overall system gain [DN/e]
E	Irradiance at detector level [W/m ²]
τ_{int}	Integration time [s]
A_{pix}	Pixel area [m ²]
DS	Dark signal [DN]

We assume three noise-components: photon noise, dark current noise, and read-noise. Both, photon and dark current noise, are Poisson distributed. The sources of read-noise are related to the sensor read out and amplifier circuits and can be described by a normal distribution with variance σ_k^2 . In this (linear) signal model the total variance σ_s^2 of the digital signal $\langle s \rangle$ is given according to the propagation of uncertainty (or propagation of error) by

$$\sigma_s^2 = \eta_{DV}^2 \cdot \eta_v^2 \cdot (\langle n_{el} \rangle + \langle n_{el}^D \rangle) + \eta_{DV}^2 \cdot \sigma_k^2. \quad (2)$$

and with Eq. (1) by

$$\sigma_s^2 = G_S \cdot \langle s \rangle + \eta_{DV}^2 \cdot \sigma_k^2 \quad (3)$$

This linear equation is a relation between variance of measured noise and averaged signal (PTC – Photon Transfer Curve).

3.2 Dark Signal Calculation

Dark signal $D = G_S \cdot \langle n_{el}^D \rangle$ is varied by changing either the integration time or the operating temperature. The shot noise component of the dark signal D is given by Eq. (4)

$$\sigma_D^2 \sim D \quad (4)$$

D is the average dark signal $D = \tau_{int} \cdot D_R$ (τ_{int} is the integration time in [s]). D_R is the average dark current rate [e⁻/s], the slope of dark signal dependence as a function of integration time (see Fig. 5).

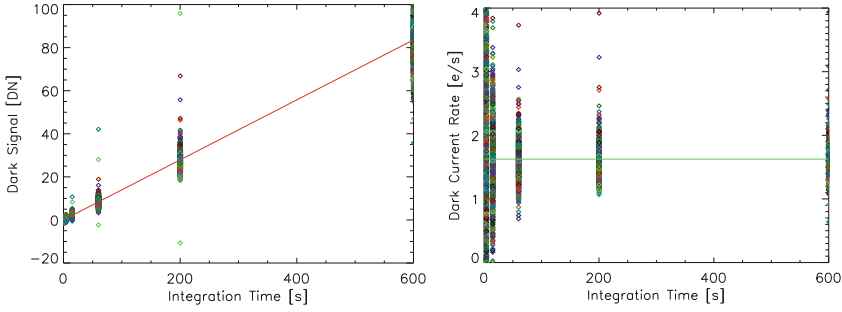


Fig. 1. Dark signal in [DN] as a function of integration time (left) and (right) dark current in [e/s] related to integration time. (Data set DS-NIR-xxxxnm-215K-160912-155847-Dark-Current-Rate, $Gain = 0.0856 \text{ V/e}$, $DC = 1.6 \text{ e/s}$, measurement at operating temperature)

Often the dark current is determined independently of the size of the detector area (5) and (6):

$$D_R [\text{nA/cm}^2] = 10^9 \cdot (Q \cdot D_R [\text{e/s}]) / A_{pix} \tag{5}$$

or the other way around (Fig. 1)

$$D_R [\text{e/s}] = D_R [\text{nA/cm}^2] \cdot 10^{-9} \frac{A_{pix}}{Q} \tag{6}$$

3.3 Temperature Dependence

We investigate the slope of dark signal dependence as a function of temperature. Different models describe this dependence. Equation (7) is an example and can be found in the book from Janesick [7]. In addition to the temperature itself, the temperature dependence of the size of the band-gap is also considered:

$$D_R = 2.55 \times 10^{15} P_A \cdot D_{FM} \cdot T^{1.5} \cdot e^{-E_g/2kT} \tag{7}$$

with

D_{FM}	Dark Current [DC] figure-of-merit @300K in [e/s] or [nA/cm ²]
$A_{pix} = 27.5 \cdot 10^{-4} \times 15 \cdot 10^{-4}$	Pixel area [cm ²]
$E_g = 1.1557 - \frac{7.021 \times 10^{-4} T^2}{1108 + T}$	Silicon bandgap energy as a function of temperature [eV]
$k = 8.6173324 \times 10^{-5}$	Boltzmann's constant [eV/K]
$Q = 1.602176565 \times 10^{-19}$	elementary charge [C] = [As]

To analyze the temperature dependence, D_{FM} has to be determined from $DC(T)$ and $E_g(T)$ at 300K. There is also an empirical equation (8) for the temperature

dependence of the dark current, which is specified by e2v for their detectors (private communication: Michael Skegg):

$$D_R = D_R(T = 300K) \cdot 122 \cdot T^3 \cdot e^{-6400/T} \tag{8}$$

A modification describes the temperature dependence better:

$$D_R = D_{FM}^* \cdot T^3 \cdot e^{-6400/T} D_{FM}^* = D_R(T = 300K) \tag{9}$$

3.4 Measurement Setup

The measurement set up is described in detail in Williges [15] and Hohn [5]. A top view of the setup is provided in the Fig. 2.

The Thermal Vacuum Chamber is equipped with one optical window with a diameter of 200 mm. Due to spatial constraints a mounting system was designed where two FPAs and the reference detector are aligned along and parallel to the optical axis, with the detectors facing the optical axis perpendicularly. A plane folding mirror mounted on a linear manipulator moving along the OA (optical axis) allowed for transfer of light to each detector subsequently. In combination with a spatial off-set of each detector with respect to the OA this allowed for equal optical path lengths, guarantying equal illumination on each detector.

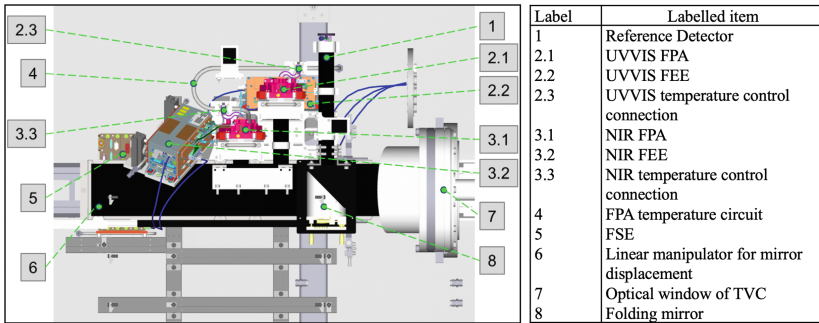


Fig. 2. CAD drawing of the opto-mechanical mounting unit developed for the Sentinel 4 verification campaign. The location of the two test detectors and the reference detector, the according electronic units as well as the cooling circuits can be seen. The folding mirror is in position 1, able to transfer light from outside of the TVC to DLR’s reference detector.

The measurement mode was Long Dark (with separate read-out of image and store areas). Long dark mode is a special mode used for this frame transfer CDD, in which the store section is read out line-by-line just before the end of the long integration period, before the frame transfer from the image area into the store are is performed. Subsequently the image area is read out in the usual manner. This method allows the determination of the dark current in the store

area under the store shield, as well as in the image area. Due to the different processing, the image and store areas may generate different dark currents.

4 Results and Discussion

The measurements of the dark current were made with an integration time of $\tau_{int} = 30\text{ s}$ with continuous cooling from 293 K to 215 K over a period of 12 h. After 10 h cooling stopped and temperatures increased again. The measurement mode was Long Dark (with separate read-out of image and store areas). All 42.723 temperature measurements for NIR and VIS detector are plotted in Fig. 3(left).

If one follows the paper of Widenhorn [14], then the dark current can be described according to the Arrhenius law:

$$D_R = D_R(T_0 = 300\text{ K}) \cdot \exp\left(-\frac{\Delta E}{kT}\right) \tag{10}$$

D_R is the dark current in [e/s] and ΔE is the activation energy.

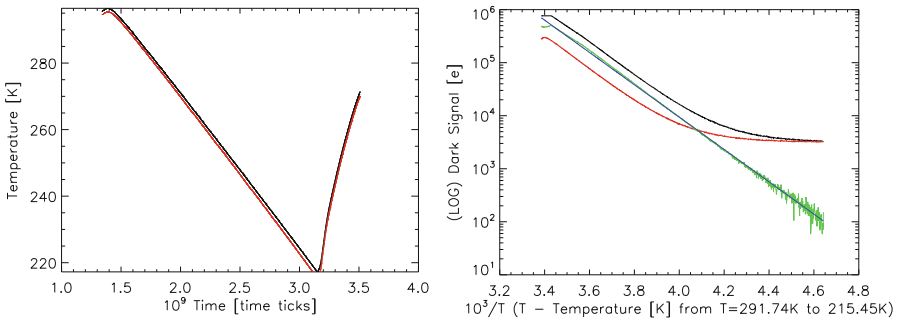


Fig. 3. Left: Temperature profile as a function of time. (black - UVVIS-Detector, red - NIR-Detector), Right: Temperature dependence of the dark signal in the image (black) and storage (red) zone, the evaluated dark signal is the difference of the signal in the image and store zone (blue) at pixel (157, 350) (Color figure online)

A first look at the data shows that the measurement reflects the expected dependency very well. In the following, the measurements shall be compared with the expected mathematical dependencies. The green line in Fig. 3(right) is the difference between the dark signal from the image zone and store zone. A fit with the exponential approach (“Arrhenius plot”, Eq. (10)) fits the experimental data quite well (see Fig. 3(right)).

The result and the comparison with the model Eqs. (7) and (9) shows Fig. 4. A look on Fig. 4(right) shows, that the relative differences derived from the plot 4(left) fits the model well between 220 K and 280 K . (The reference temperature is chosen at $T = 276.6\text{ K}$.) At about 280 K , a significant kink is visible. Between

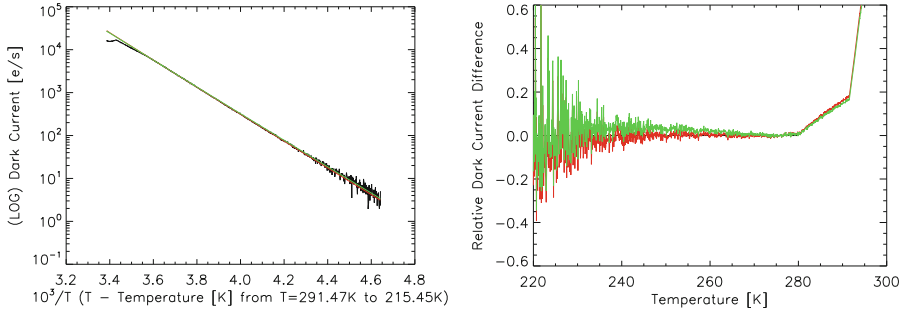


Fig. 4. Left: Temperature profile as a function of temperature according to the Arrhenius plot. (black - DC from NIR-Detector, red - model based on Eq. (7), green - model based on Eq. (9)) Right: Temperature dependence of the relative differences of the dark signal and model signal at pixel (157, 350). (Color figure online)

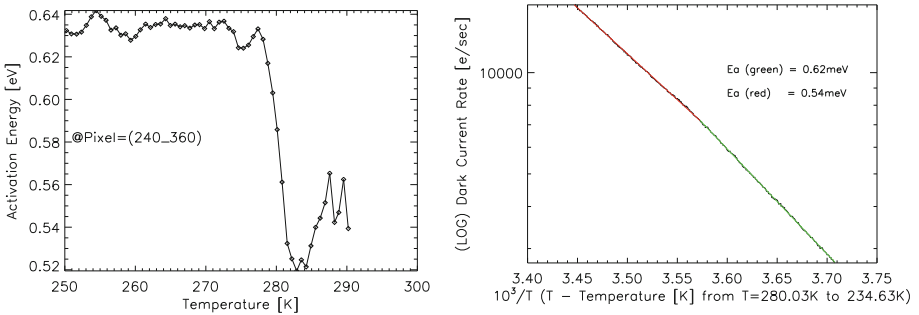


Fig. 5. NIR temperature dependence of the dark current rate @Pixel=(240, 360). Noticeable is the significant change in the activation energy at 280K.

280K and 295K one will find a linear deviation from the model. Above 295K, the deviation is due to the saturation of the signal.

In the following, the kink will be examined more closely at about 280K. For this purpose, the increase should be considered as a function of temperature. Figure 5(left) shows the result. Here it becomes clear that at $T = 280K$, the activation energy changes significantly. Figure 5(right) shows the plot with the fit for the two different climbs.

The spatial distribution of the activation energy is shown in Fig. 6. Obviously, the standard deviation is very small (1%–2%) and there are no abnormalities in the spatial distribution of the two activation series.

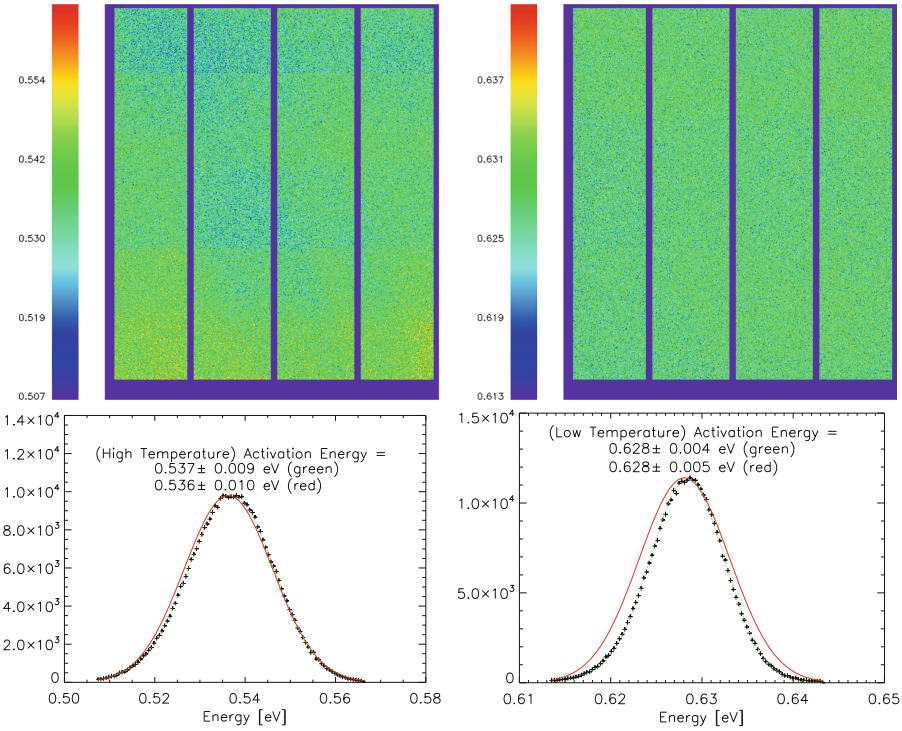


Fig. 6. Spatial distribution and histogram of the activation energy. (Histogram values (+), in green a Gauss-fit to the histogram values, red line is the Gauss distribution derived from mean and standard-deviation of the histogram) (Color figure online)

Finally, the dark current should be calculated at the point where the low temperature and the high temperature dependence intersect. Temperature and dark current are calculated according to Eq. (11)

$$\begin{aligned}
 T_C &= \frac{\Delta E^{HT} - \Delta E^{LT}}{k(\log D_{R0}^{HT} - \log D_{R0}^{LT})} \\
 D_R(T_C) &= D_{R0}^{LT} \cdot \exp\left(-\frac{\Delta E^{LT}}{kT_C}\right)
 \end{aligned}
 \tag{11}$$

The histogram of the crossing Temperature and the dark current at T_C is shown in Fig. 7.

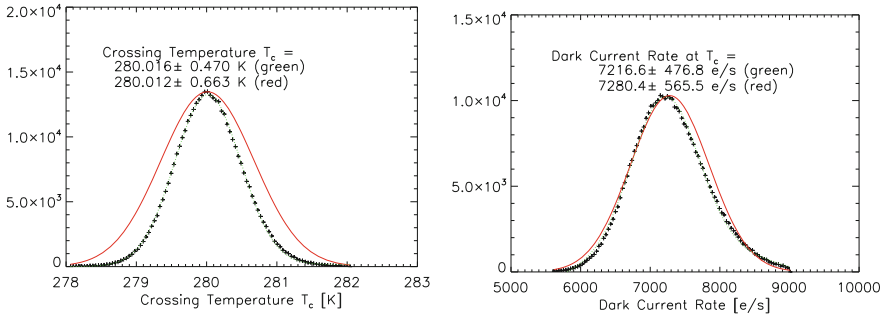


Fig. 7. Histogram of the crossing temperature and the dark current at T_c

5 Conclusion and Outlook

In this paper we report about an investigation of temperature behaviour of the dark signal/current between 215K and 290K for a scientific CCD device. The data set has an exceptionally high number of measuring points (900 measurements with an integration time of 30s). The temperature measurements are synchronous to the dark signal measurements.

Basically, the temperature behavior of the dark current is as expected theoretically and experimentally (see Eqs. (7) and (9)).

Two parts of the temperature dependence of the dark current could be detected. These can be characterized, for example, by different activation energies. There is a clear separation of the activation energies at 280K. The result is

- Low temp activation energy = 0.628 ± 0.004 eV
- High temp activation energy = 0.537 ± 0.009 eV
- The variation width for the activation energies for the entire chip was extremely small (1%–2%).

Due to this, no variation of the pre-factor with the activation energy was found (Meyer-Neldel rule). (See Fig. 3 from the paper [14].)

The main finding of the article is the change of the dark current rate with the temperature. The reason is assumed to be the changing electric field within the silicon chip, since the depletion depth decreases as the number of electrons increases. Further measurements, which are to be carried out as part of the verification of the flight model, should deepen this knowledge.

Acknowledgments. The presented work has been performed under ESA contract. The authors would like to express their thanks to their respective colleagues at Airbus Defense and Space, ESA and EUMETSAT, and to all partner companies within the Sentinel 4 industrial consortium for their valuable contributions to the continuing success of this very challenging program. This article has been produced with financial assistance of the EU. The views expressed herein shall not be taken to reflect

the official opinion of the EU. The work presented in this paper was supported by a large number of people. The authors wish to acknowledge Max Harlander, Markus Hermesen, Yves Levillain, Christian Williges, Hannes Roßmann and Stefan Hilbert, for their exceptional support.

References

1. Candeias, H., et al.: RTS effect detection in Sentinel-4 data. In: Earth Observing Systems XXII, vol. 10402, p. 104021B. International Society for Optics and Photonics (2017)
2. Eckardt, A., Krutz, D., Säuberlich, T., Venus, H., Reulke, R.: Verification and calibration of the DESIS detector. In: Imaging Spectrometry XXII: Applications, Sensors, and Processing, vol. 10768, p. 107680B. International Society for Optics and Photonics (2018)
3. Hermesen, M., Hohn, R., Skegg, M., Woffinden, C., Reulke, R.: Realization of the electrical Sentinel 4 detector integration. In: Earth Observing Systems XXII, vol. 10402, p. 1040219. International Society for Optics and Photonics (2017)
4. Hinger, J., Hohn, R., Gebhardt, E., Reichardt, J.: The Sentinel-4 UVN focal plane assemblies. In: Earth Observing Systems XXII, vol. 10402, p. 1040217. International Society for Optics and Photonics (2017)
5. Hohn, R., Skegg, M.P., Hermesen, M., Hinger, J., Williges, C., Reulke, R.: The S4 focal plane subsystem. In: Earth Observing Systems XXII, vol. 10402, p. 1040216. International Society for Optics and Photonics (2017)
6. Janesick, J.R.: Scientific Charge-Coupled Devices. SPIE Press Book, Bellingham (2001)
7. Janesick, J.R.: Photon Transfer. SPIE Press Book, Bellingham (2007)
8. Kopeika, N.S.: A System Engineering Approach to Imaging. SPIE Press Book, Bellingham (1998)
9. Krutz, D., et al.: The instrument design of the DLR earth sensing imaging spectrometer (DESI). *Sensors* **19**(7), 1622 (2019)
10. Popowicz, A.: Analysis of dark current in BRITE nanostellite CCD sensors. *Sensors* **18**(2), 479 (2018)
11. Schwarzer, H., Eckardt, A., Reulke, R.: Verification of a spectrometer breadboard for characterization of a future spaceborne sensor. In: Huang, F., Sugimoto, A. (eds.) PSIVT 2015. LNCS, vol. 9555, pp. 285–295. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-30285-0_23
12. Skegg, M.P., et al.: The Sentinel-4 detectors: architecture and performance. In: Earth Observing Systems XXII, vol. 10402, p. 1040218. International Society for Optics and Photonics (2017)
13. Swindells, I., Morris, D.: Electro-optical sensors for Earth observation missions. In: Butler, J.J., Xiong, X.J., Gu, X. (eds.) Earth Observing Systems XXIII, vol. 10764, pp. 56–68. International Society for Optics and Photonics, SPIE (2018). <https://doi.org/10.1117/12.2321186>
14. Widenhorn, R., Blouke, M.M., Weber, A., Rest, A., Bodegom, E.: Temperature dependence of dark current in a CCD. In: Sensors and Camera Systems for Scientific, Industrial, and Digital Photography Applications III, vol. 4669, pp. 193–201. International Society for Optics and Photonics (2002)
15. Williges, C., et al.: Verification of the Sentinel-4 focal plane subsystem. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **42** (2017)



An Extended Stochastic Cloning Method for Fusing Multi-relative Measurements

Hongmou Zhang^(✉), Dirk Baumbach, Denis Griebbach, and Anko Börner

German Aerospace Center, Rutherfordstr. 2, 12489 Berlin, Germany
Hongmou.Zhang@dlr.de

Abstract. One of the most important tasks for visual inertial odometry systems is pose estimation. By integrating system poses, motion trajectory of the system can be obtained. Due to errors existing in calculations, the accumulated errors grow unbounded. To decrease the drift, keyframes and loop-closure information can be used as additional references for the system. To use this information, the system should be able to handle multi-relative measurements which cross many periods of filter cycles. In Kalman filter based system the fusion of such information is one of the toughest problems. In this paper, we propose an extended stochastic cloning method to overcome this problem. The proposed method is based on the error state Kalman filter. It also can be used in other Kalman filters. The experimental results show that based on the proposed method trajectory errors and uncertainties of filtered results are decreased significantly. At the same time, the IMU's biases are modeled as a random-walk noise and be updated as well. This way, by using keyframes and loop-closure information, the proposed method is able to improve the accuracy of the sensor models.

Keywords: Kalman filter · Stochastic cloning · Visual odometry · Measurement fusion

1 Introduction

Ego-motion estimation is one of the key research topics for applications like robotics, autonomous driving etc. A common way to obtain ego-motion information is by using vision-aided inertial navigation system, which is an integrated method of camera and *inertial measurement unit* (IMU). A relative motion to the previous measurement can be obtained by fusing data from camera and IMU. Since noise and errors exist in each measurement, the overall motion trajectory which is obtained by only integrating relative motions subjects to large errors. Post processing steps must be applied to eliminate errors and finally output an optimized motion trajectory.

In computer vision community, the Kalman filter based method [1] is one of the famous post processing methods. Beside Kalman filter, many methods [2–4] are widely used as well. Some papers claim that optimization-based methods

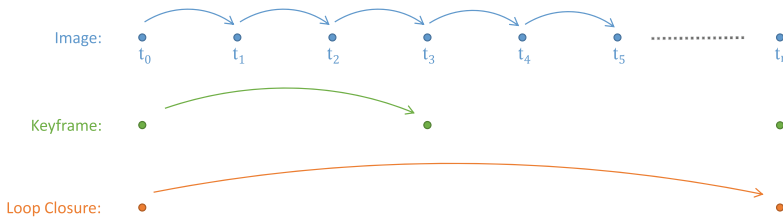


Fig. 1. Illustration of the relationship between standard image frames, keyframes, and loop-closure frames. Easy to see that all of them measure a transformation between two frames.

outperform filter based method [5,6]. However, Kalman filter based method has its advantage. It requires less computation resources, and it is able to provide the quality of the estimation. At the same time, the performance gap between Kalman filter and other methods is not significant according to our test. Therefore, our research focuses on the Kalman filter based ego-motion estimation method.

With each new relative motion estimation arriving, system state is updated by fusing the data into Kalman filter. Such frame-by-frame pose calculation method is commonly used for visual odometry. However, due to imperfect calibration and noise naturally exist in every sensor, errors cannot be completely eliminated by filter. As a result, the accumulated errors grow unlimited.

To overcome above mentioned problem, keyframe [7,8] and loop-closure method [9] are introduced for visual odometry. Both ideas involve previous measurements as additional constraints to optimize current system state. By the former method, most representative frames are selected as keyframes, and several normal frames exist in between keyframe pairs. Once a new keyframe is obtained, frame-by-frame estimated trajectory is optimized again by including only the new keyframe and its adjacent keyframes. By loop closure method, system detects whether a current environment is the same as in a previously already passed scene. Accumulated error can be eliminated by using successfully detected loop-closure information. Figure 1 shows relationship between standard image frames, keyframes, and loop-closure frames.

Keyframe and loop-closure method perform well in optimization-based visual odometry systems. To involve these methods, previous measurements must be kept by system. Such requirement makes them difficult to be included in Kalman filter based method which does not keep any historical data other than the one previous state. We are inspired by [10] and propose a novel method, which can handle multi-historical states in Kalman filter. With the proposed method, keyframe and loop closure information are easily involved into Kalman filter based system. Moreover, by using this information, uncertainties of filtered results and IMU's biases decrease significantly. This way, keyframes and loop closure information are used to improve the accuracy of the sensor models.

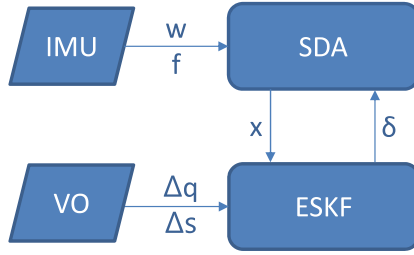


Fig. 2. Data fusion of *visual odometry* (VO) and IMU using *strapdown algorithm* (SDA) and ESKF.

This paper is organized as follows: In Sect. 2, an introduction to *error state Kalman filter* (ESKF) is given. ESKF is suitable for vision-aided inertial systems, therefore, the proposed method is based on ESKF but also can be easily used in other Kalman filters. In Sect. 3 the proposed extended stochastic cloning method is described. Experimental results are presented in Sect. 4, and Sect. 5 concludes the paper.

2 Error State Kalman Filter

Just by mathematical integration of raw data provided by an IMU through the *strapdown algorithm* (SDA) it is possible to conclude on an absolute position or orientation in space. Unfortunately, the IMU output data (angular rate \boldsymbol{w} , acceleration \boldsymbol{f}) is associated with various sensor errors such as scaling, bias, non-linearity etc. so that the exclusive use of an IMU to compute an absolute pose will quickly lead to unacceptable results. By adding additional information such as from *visual odometry* (VO) the importance of these sensor errors can be reduced and the faulty navigation solution can be corrected. This sensor data fusion (Fig. 2) is realized with an ESKF. It does not directly estimate the variables of interest (\boldsymbol{x}), but their errors ($\boldsymbol{\delta}$), which in turn serve as a correction for the strapdown solution.

2.1 Strapdown Algorithm

The state vector of the strapdown algorithm (1) contains the rotation from body to navigation frame represented as a quaternion \boldsymbol{q}_b^n , the position \boldsymbol{s}^n and speed \boldsymbol{v}^n defined in the navigation frame as well as the body-fixed offsets of the IMU angular rate \boldsymbol{b}_w or acceleration sensor \boldsymbol{b}_f . Each of the physical quantities is subdivided into 3 spatial axes, resulting in a state vector length of 16.

$$\boldsymbol{x}_{16 \times 1} = [\boldsymbol{q}_b^n \ \boldsymbol{s}^n \ \boldsymbol{v}^n \ \boldsymbol{b}_w \ \boldsymbol{b}_f]^T \tag{1}$$

The Eqs. (2) to (6) describe the time-discrete propagation of the individual state variables for the time step $dt = t_{k+1} - t_k$. The operator \circ in (2) describes a

quaternion multiplication between the previous $\mathbf{q}_{b,k}^n$ and an orientation change, fed from the offset-corrected IMU angular rate measurement $\mathbf{w}_k = \tilde{\mathbf{w}}_k - \mathbf{b}_{w,k}$. The position \mathbf{s}_k^n is propagated with the trapezoidal rule. The velocity update (4) is calculated with the offset-corrected acceleration $\mathbf{f}_k = \tilde{\mathbf{f}}_k - \mathbf{b}_{f,k}$. The cross product inside the second bracket describes a rotation correction, \mathbf{g}^n stands for the local gravity vector. Both IMU offsets (5, 6) do not change during a time step.

$$\mathbf{q}_{b,k+1}^n = \mathbf{q}_{b,k}^n \circ \mathbf{q}(\mathbf{w}_k \cdot dt) \tag{2}$$

$$\mathbf{s}_{k+1}^n = \mathbf{s}_k^n + (\mathbf{v}_k^n + \mathbf{v}_{k+1}^n) \cdot dt/2 \tag{3}$$

$$\mathbf{v}_{k+1}^n = \mathbf{v}_k^n + \mathbf{R}(\mathbf{q}_{b,k}^n) \cdot (\mathbf{f}_k + 0.5 \cdot \mathbf{w}_k \times \mathbf{f}_k) \cdot dt - \mathbf{g}^n \cdot dt \tag{4}$$

$$\mathbf{b}_{w,k+1} = \mathbf{b}_{w,k} \tag{5}$$

$$\mathbf{b}_{f,k+1} = \mathbf{b}_{f,k} \tag{6}$$

2.2 Error State Equations

The error state Kalman filter estimates the error values of the physical quantities to the navigation system. Upon arrival of visual odometry measurement, the navigation system pose error is cloned ($\delta\alpha_c, \delta\mathbf{s}_c$) and the error state vector (7) is extended by it, which yields to a δ vector length of 21. This cloning serves to accurately project the system uncertainties between two visual odometry surveys.

$$\delta_{21 \times 1} = [\delta\alpha \ \delta\mathbf{s} \ \delta\mathbf{v} \ \delta\mathbf{b}_w \ \delta\mathbf{b}_f | \delta\alpha_c \ \delta\mathbf{s}_c]^T \tag{7}$$

$$\delta_{k+1} = \Phi_k \cdot \delta_k + \mathbf{G}_k \cdot \mathbf{w}_k \tag{8}$$

$$= \begin{pmatrix} (\mathbf{I}_{15} + \mathbf{F}_{15} \cdot dt) & \mathbf{0}_{15 \times 6} \\ \mathbf{0}_{6 \times 15} & \mathbf{I}_6 \end{pmatrix} \cdot \delta_k + \begin{pmatrix} \mathbf{G}_{15 \times 12} \cdot dt \\ \mathbf{0}_{6 \times 12} \end{pmatrix} \cdot \mathbf{w}_k \tag{9}$$

$$\mathbf{F}_{15} = \begin{pmatrix} \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & -\mathbf{R} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ -[\mathbf{R}\mathbf{f}_k]_{\times} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & -\mathbf{R} \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{pmatrix} \tag{10}$$

To propagate the errors δ in (8) the time-discrete transition matrix Φ_k is obtained from the continuous process matrix \mathbf{F}_{15} by first order Taylor series approximation, where \mathbf{F}_{15} is obtained by linearizing the strapdown equations in Sect. 2.1. The process noise \mathbf{w}_k in (8) is largely determined by the properties of the IMU used. This vector consists of the standard deviations for the angular rate noise \mathbf{n}_w , the angular rate bias instability \mathbf{n}_{b_w} , the acceleration sensor noise \mathbf{n}_f and the acceleration bias instability \mathbf{n}_{b_f} .

$$\mathbf{w}_{12 \times 1} = [\mathbf{n}_w \ \mathbf{n}_{b_w} \ \mathbf{n}_f \ \mathbf{n}_{b_f}]^T \tag{11}$$

The associated time-discrete process noise matrix \mathbf{G} can be set up as follows:

$$\mathbf{G}_{15 \times 12} = \begin{pmatrix} \mathbf{R} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{R} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{pmatrix} \quad (12)$$

2.3 Visual Odometry Observation

The visual odometry measurement from the stereo camera system $\mathbf{T}_{k,vo}^{k+1}$ provides 6 dof relative pose information between two time steps t_k and t_{k+1} . The relative pose is composed of 3d angle and position information represented as a homogeneous transformation matrix:

$$\mathbf{T}_{k,vo}^{k+1} = \mathbf{T}(\Delta\mathbf{q}, \Delta\mathbf{s})_{vo} = \begin{pmatrix} \mathbf{R}(\Delta\mathbf{q}) [\Delta s_x \ \Delta s_y \ \Delta s_z]^T \\ \mathbf{0}_{1 \times 3} & & 1 \end{pmatrix}_{vo} \quad (13)$$

The projection of (13) on the absolute state vector (1) can be calculated by the difference between the strapdown algorithm poses at time t_k and t_{k+1} .

$$\mathbf{T}_{k,n}^{k+1} = \mathbf{T}(\mathbf{q}_k, \mathbf{s}_k)_n^{-1} \cdot \mathbf{T}(\mathbf{q}_{k+1}, \mathbf{s}_{k+1})_n \quad (14)$$

The navigation filter estimates error terms, therefore its observation data is calculated from the difference between the strapdown algorithm and the visual odometry. If a VO reading is available, the error measurement $\Delta\mathbf{T}_k^{k+1}$ can be described by:

$$\Delta\mathbf{T}_{k,n}^{k+1} = \mathbf{T}_{vo}^n \left(\mathbf{T}_{k,vo}^{k+1} \right)^{-1} \left(\mathbf{T}_{vo}^n \right)^{-1} \mathbf{T}_{k,n}^{k+1} \quad (15)$$

\mathbf{T}_{vo}^n transforms measurement from VO frame to IMU frame. By partial derivation of (14) towards the angle and position errors at the current ($\delta\boldsymbol{\alpha}$, $\delta\mathbf{s}$) or previous time ($\delta\boldsymbol{\alpha}_c$, $\delta\mathbf{s}_c$), the measurement matrix \mathbf{H}_{k+1} can be set up.

$$\mathbf{H}_{k+1} = \left(\mathbf{H}_{k+1|k} \ \mathbf{0}_{6 \times 9} \ \mathbf{H}_{k|k} \right) \quad (16)$$

$\mathbf{H}_{k+1|k}$ is defined in as follows:

$$\mathbf{H}_{k+1|k} = \frac{\partial \mathbf{T}_{k,n}^{k+1}}{\partial (\delta\boldsymbol{\alpha}_{k+1}, \delta\mathbf{s}_{k+1})} \quad (17)$$

According to [10], it is known that

$$\mathbf{H}_{k|k} = -\mathbf{H}_{k+1|k} \mathcal{F} \quad (18)$$

\mathcal{F} is the accumulative transformation matrix, where $\mathcal{F} = \prod_{i=1}^m \boldsymbol{\Phi}_{k+i}$.

2.4 Strapdown Correction

After a successful filter cycle, the error filter state vector δ is used to enhance the state vector \mathbf{x} of the strapdown algorithm. Since visual odometry provides a measurement with 6 degrees of freedom, i.e. angle and position are not processed separately from each other, the pose is also corrected in the form of a multiplication with homogeneous transformation matrices.

$$\mathbf{T}(\mathbf{q}_{b,k+1}^{n,+}, \mathbf{s}_{k+1}^{n,+}) = \mathbf{T}(\delta\boldsymbol{\alpha}, \delta\mathbf{s})^{-1} \cdot \mathbf{T}(\mathbf{q}_{b,k+1}^{n,-}, \mathbf{s}_{k+1}^{n,-}) \tag{19}$$

The remainder of the state vector (1) is corrected by simply subtracting the associated error terms of (7).

$$\mathbf{v}_{k+1}^+ = \mathbf{v}_{k+1}^- - \delta\mathbf{v} \tag{20}$$

$$\mathbf{b}_{w,k+1}^+ = \mathbf{b}_{w,k+1}^- - \delta\mathbf{b}_w \tag{21}$$

$$\mathbf{b}_{f,k+1}^+ = \mathbf{b}_{f,k+1}^- - \delta\mathbf{b}_f \tag{22}$$

3 Extended Stochastic Cloning

Relative motion estimation from camera system is an interdependent measurement of the system state at two time instants. The stochastic cloning method is commonly used to processing such measurement. As shown in Fig. 1, the relative motion estimation of keyframe and loop-closure are interdependent measurements as well. In this case, the system states at every keyframe time instants must be kept as well, this means that Kalman filter must be able to handle multiple system states simultaneously. An extended stochastic cloning method is proposed for such purpose.

First of all, instead of the standard stochastic cloning, the error state is augmented as follows:

$$\Delta\check{\mathbf{x}}_k = (\Delta\mathbf{x}_k, \Delta\mathbf{x}_{e,k}, \Delta\mathbf{x}_{c,k})^T \tag{23}$$

$\Delta\mathbf{x}_k$ is the updated system state at time k , $\Delta\mathbf{x}_{e,k}$ and $\Delta\mathbf{x}_{c,k}$ are augmented system states for keyframe and standard frame. The keyframe term $\Delta\mathbf{x}_{e,k}$ must be located on the left of the standard-frame term $\Delta\mathbf{x}_{c,k}$. Therefore, the covariance matrix equals

$$\check{\mathbf{P}}_k = \left[\begin{array}{cc|c} \mathbf{P}_k & \mathbf{P}_{e,k} & \mathbf{P}_{c,k} \\ \mathbf{P}_{e,k} & \mathbf{P}_{e,k} & \mathbf{0} \\ \hline \mathbf{P}_{c,k} & \mathbf{0} & \mathbf{P}_{c,k} \end{array} \right] \tag{24}$$

In Eq. (24), $\mathbf{P}_k = \mathbf{P}_{e,k} = \mathbf{P}_{c,k}$. The reason for the different notation is to indicate explicitly the covariance elements for each term in the error state.

The zero terms in the covariance matrix are cross covariance between the cloned terms for standard frames and keyframes. Because the standard frames and keyframes are independently measured by different modules, therefore, they are uncorrelated.

Next, the prediction steps are performed whenever each IMU measurement arrives.

$$\check{\mathbf{P}}_{k+n|k} = \check{\mathbf{\Phi}}_{k+n} \check{\mathbf{P}}_k \check{\mathbf{\Phi}}_{k+n}^\top + \check{\mathbf{G}}_{k+n} \mathbf{Q}_{k+n} \check{\mathbf{G}}_{k+n}^\top \quad (25)$$

$\check{\mathbf{\Phi}}_{k+n}$ is the augmented state transformation matrix. \mathbf{Q}_{k+n} is the covariance matrix of the process noise. $\check{\mathbf{G}}_{k+n}$ is the augmented matrix of the noise transition matrix \mathbf{G}_{k+n} in Eq. (12):

$$\check{\mathbf{G}}_{k+n} = \begin{bmatrix} \mathbf{G}_{k+n} \\ \mathbf{0} \end{bmatrix} \quad (26)$$

Once the next standard frame is obtained, the filter state and the covariances are updated. Note that the top-left block in Eq. (24) is an independent entity. Therefore, in the propagated covariance matrix $\check{\mathbf{P}}_{k+n|k}$, the top-left block is entirely updated.

Consider that a new keyframe is detected at time $k+m$. The system state can be updated as following

$$\Delta \check{\mathbf{x}}_{k+m} = \Delta \check{\mathbf{x}}_k + \check{\mathbf{K}} \mathbf{q}_i \quad (27)$$

$$\check{\mathbf{P}}_{k+m|k+m} = \check{\mathbf{P}}_{k+m|k} - \check{\mathbf{K}} \check{\mathbf{H}}_{k+m} \check{\mathbf{P}}_{k+m|k} \quad (28)$$

$\check{\mathbf{K}}$ is Kalman gain which equals

$$\check{\mathbf{K}} = \check{\mathbf{P}}_{k+m|k} \check{\mathbf{H}}_{k+m}^\top \check{\mathbf{S}}^{-1} \quad (29)$$

where

$$\check{\mathbf{S}} = \check{\mathbf{H}}_{k+m} \check{\mathbf{P}}_{k+m|k} \check{\mathbf{H}}_{k+m}^\top + \Sigma_{\mathbf{q}_i} \quad (30)$$

$\Sigma_{\mathbf{q}_i}$ is the covariance matrix of the innovation vector \mathbf{q}_i which includes three Euler angles and three translation elements corresponding to \mathbf{T}_i . $\check{\mathbf{H}}_{k+m}$ is the augmented observation matrix which is defined as follows:

$$\check{\mathbf{H}}_{k+m} = [\mathbf{H}_{k+m|k}, \mathbf{0}_{6 \times 9}, \mathbf{H}_{k|k}, \mathbf{0}_{6 \times 9}, \mathbf{0}_{6 \times 15}] \quad (31)$$

Here, $\mathbf{0}_{6 \times 15}$ is the entry relative to the standard frame, $\mathbf{H}_{k|k}$ relates to the cloned term in the augmented error state $\Delta \check{\mathbf{x}}_k$, and $\mathbf{H}_{k+m|k}$ relates to the original terms in $\Delta \check{\mathbf{x}}_k$. $\mathbf{0}_{6 \times 9}$ relates to the IMU terms, because the camera measurement does not provide any additional information about the IMU entries in $\Delta \check{\mathbf{x}}_k$. $\mathbf{H}_{k+m|k}$ is defined as follows:

$$\mathbf{H}_{k+m|k} = \frac{\partial \mathbf{T}_{k,n}^{k+m}}{\partial (\delta \boldsymbol{\alpha}_{k+m}, \delta \mathbf{s}_{k+m})} \quad (32)$$

$\mathbf{H}_{k|k}$ is the observation matrix relative to the keyframe. Because the error measurement is derived as follows:

$$\Delta \mathbf{T}_{k,n}^{k+m} = \mathbf{T}_{vo}^n \left(\mathbf{T}_{k,vo}^{k+m} \right)^{-1} \left(\mathbf{T}_{vo}^n \right)^{-1} \mathbf{T}_{k,n}^{k+m} \quad (33)$$

Therefore, $\mathbf{H}_{k|k}$ can be obtained with partial derivatives of $\mathbf{T}_{k,n}^{k+m}$ w.r.t. $(\delta\alpha_k, \delta\mathbf{s}_k)$ (see Eq. 34). In this way, the accumulative transformation matrix (Eq. 10) is not needed anymore; this is even faster than the standard stochastic cloning:

$$\mathbf{H}_{k|k} = \frac{\partial \mathbf{T}_{k,n}^{k+m}}{\partial (\delta\alpha_k, \delta\mathbf{s}_k)} \quad (34)$$

Back to Eq. (27), because the elements in $\Delta\check{\mathbf{x}}_k$ are kept as being zero, therefore, $\Delta\check{\mathbf{x}}_{k+m} = \check{\mathbf{K}}\mathbf{q}_i$. Finally, the updated error state $\Delta\check{\mathbf{x}}_{k+m}$ is added to the predicted state $\tilde{\mathbf{x}}_{k+m}$ for obtaining the filtered system state

$$\mathbf{x}_{k+m} = \tilde{\mathbf{x}}_{k+m} + \Delta\mathbf{x}_{k+m} \quad (35)$$

$\Delta\mathbf{x}_{k+m}$ is a vector which includes the original state terms in $\Delta\check{\mathbf{x}}_{k+m}$. In this way, the filter is updated by using keyframe measurements. All elements in the system state \mathbf{x} benefit from additional keyframes and loop-closure measurements.

The above steps are repeated for subsequent measurements, the size of the filter state grows constantly. Once the number of terms in the filter state reaches a maximum, some terms must be removed.

Time-stamp intervals between keyframes are checked. A keyframe is removed if it is defined by a minimum-length interval to its neighbors. This scheme guarantees a uniform distribution of the remaining keyframes with respect to the time dimension.

4 Experimental Results

To evaluate performance of the proposed keyframe and loop-closure methods, several realistic indoor and outdoor datasets are recorded. Our test platform is named as *Integrated Positioning System* (IPS), which is an ESKF based visual inertial odometry system.

Each of the tests includes two basic steps. First, the performance of the original system is checked on the datasets. Then, the keyframe and loop-closure measurements are involved to the IPS system by using the proposed method, the complete processing is performed again.

By comparing the results, an improvement of the trajectory, of the system uncertainties, and the sensor-noise model can be easily seen.

First, a static dataset is used, the dataset is composed of a static image sequence for about 34 min in 10 Hz.

The static case is no challenge for keyframe detection. The keyframes can even be defined by a fixed time interval. However, this dataset is an ideal resource to check the correctness of the filter algorithm. Only if the proposed filter algorithm works fine, then the impact of the keyframe method to the system can be analyzed. On the other hand, as mentioned before, the loop-closure measurement is fused in the same manner as for the keyframes. Therefore, the static test is selected as the first experiment.

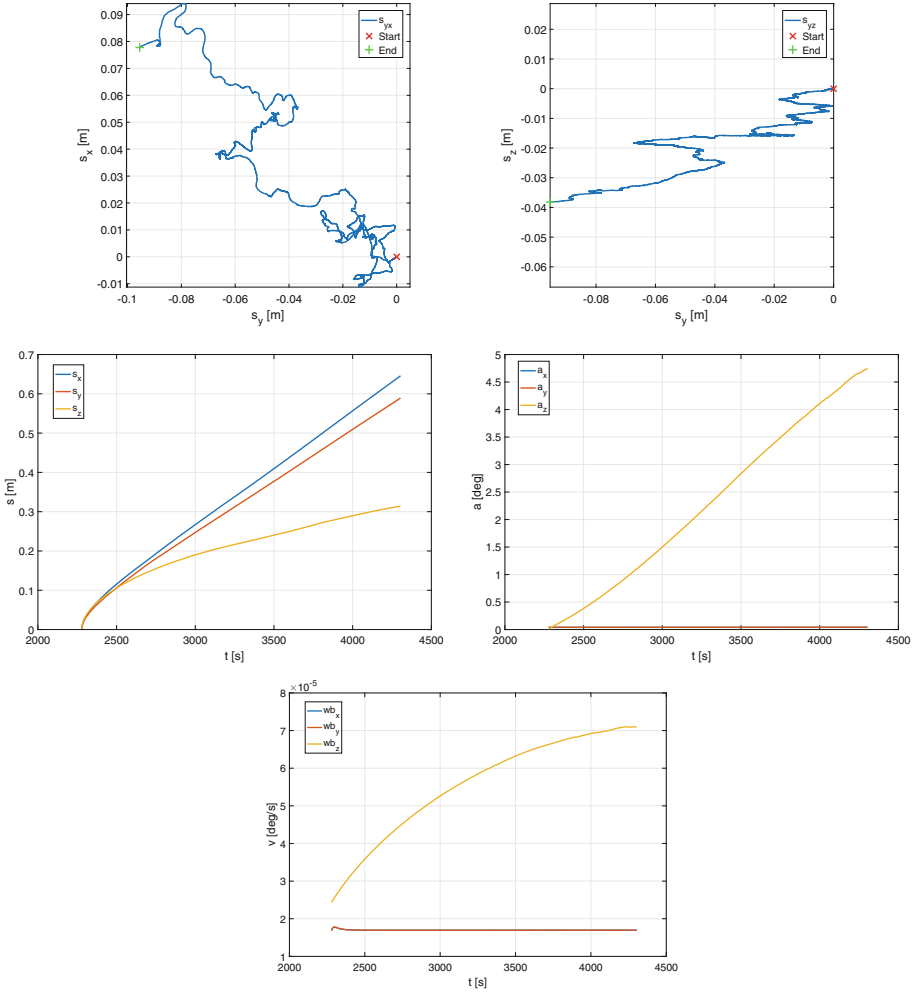


Fig. 3. *Top-left:* yx -view of the measured trajectory. *Top-right:* yz -view. *Middle-left:* Uncertainties of measured positions show that they grow over time. The horizontal axis is the time stamp. *Middle-right:* Uncertainties of measured rotations. Rotations about the x and y axis are corrected by gravity which is measured by the IMU. Therefore, rotation uncertainties for these two axes are kept in a stable range and they are overlaid in the figure. *Bottom:* Uncertainties of angular-velocity biases. these curves indicate the confidence of the modeled IMU drift.

The quality of the IPS measurements is affected by the sensor noise, any imperfect calibration, and so forth. On the other hand, although the scene is fixed in the static dataset, illumination and shadows of objects change with the process over time. Therefore, even given that the system is fixed at a physical position, the measured trajectory has a random drift.

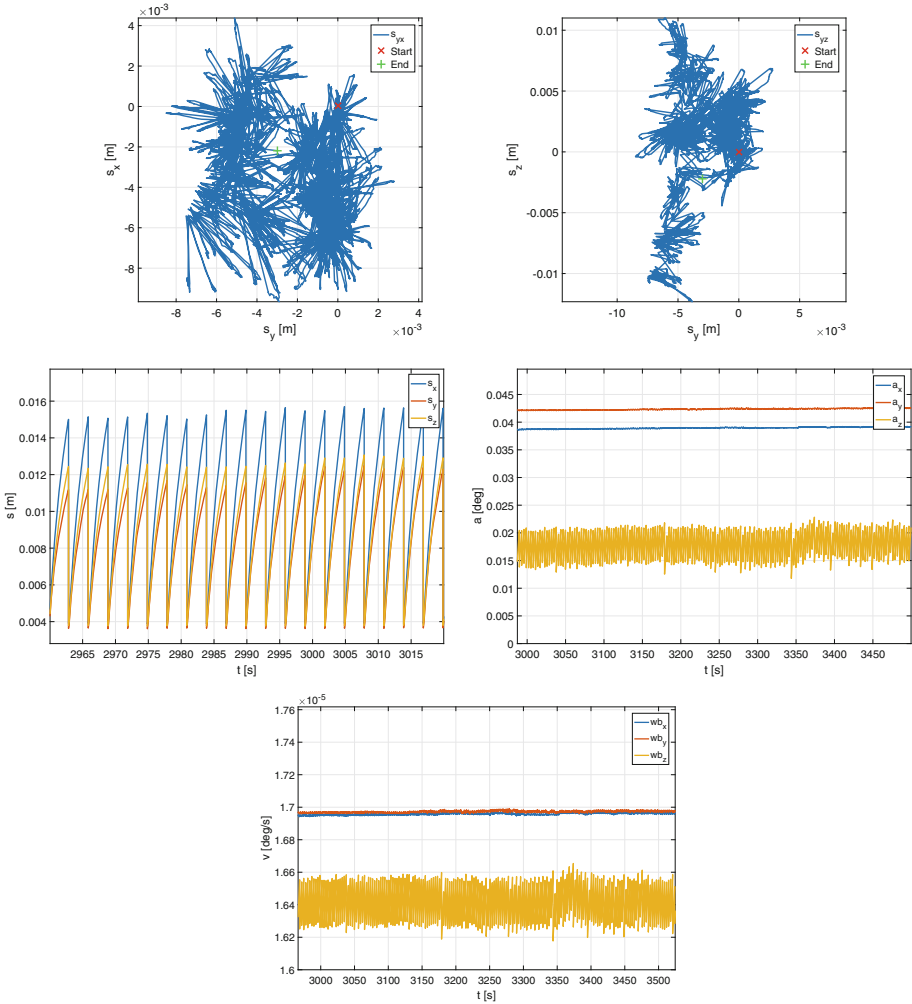


Fig. 4. *Top-left:* yx -view of measured trajectory. *Top-right:* yz -view of measured trajectory. *Middle-left:* Part of uncertainties of measured positions, uncertainties decrease once keyframe measurement arrives. Patterns repeat during the whole process. The full figure is too dense to be displayed. *Middle-right:* Part of uncertainties of measured rotation. As mentioned before, the rotation about the x and y axis is corrected by gravity, thus, there is no significant effect on both. *Bottom:* Part of uncertainties of angular velocity biases.

Figure 3 shows the measurements using the original IPS. In the test, the root mean square (RMS) 3D error between start and end point equals 0.129 m.

Next, keyframe measurements are involved into the system, the maximum keyframe interval equals 50 frames. The testing results are shown in Fig. 4. In this case, the RMS 3D error equals 0.004 m.

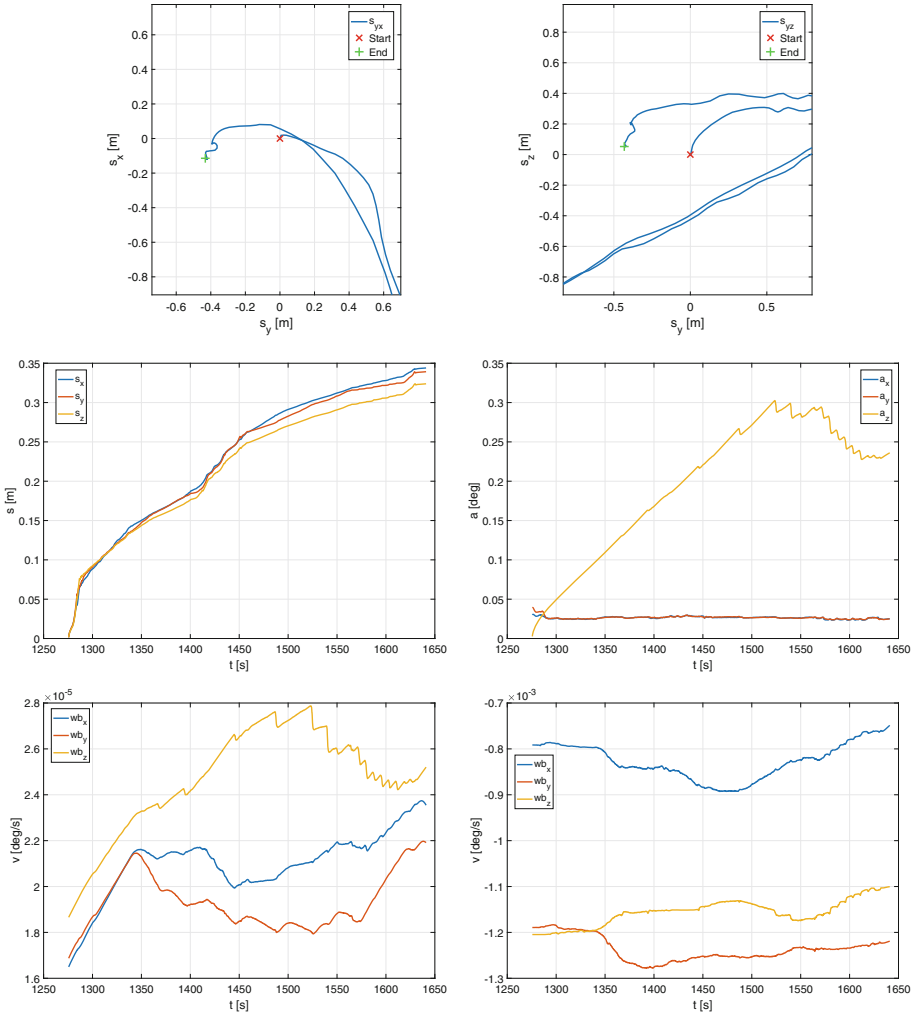


Fig. 5. *Top:* Both figures show the part of the measured trajectory. They clearly indicate an error between start and end point. *Middle-left:* Uncertainties of measured positions. *Middle-right:* Uncertainties of measured rotation. *Bottom-left:* Uncertainties of modeled angular velocity biases. *Bottom-right:* Modeled angular velocity biases.

The test shows that the filter works as expected. The error of the measured trajectory (when using the new system) is about 32 times less than for the original IPS.

Next, the system is tested on a dynamic dataset. Figure 5 shows the measurements of the original IPS. Figure 6 shows the results of the IPS with keyframe and loop-closure measurements.

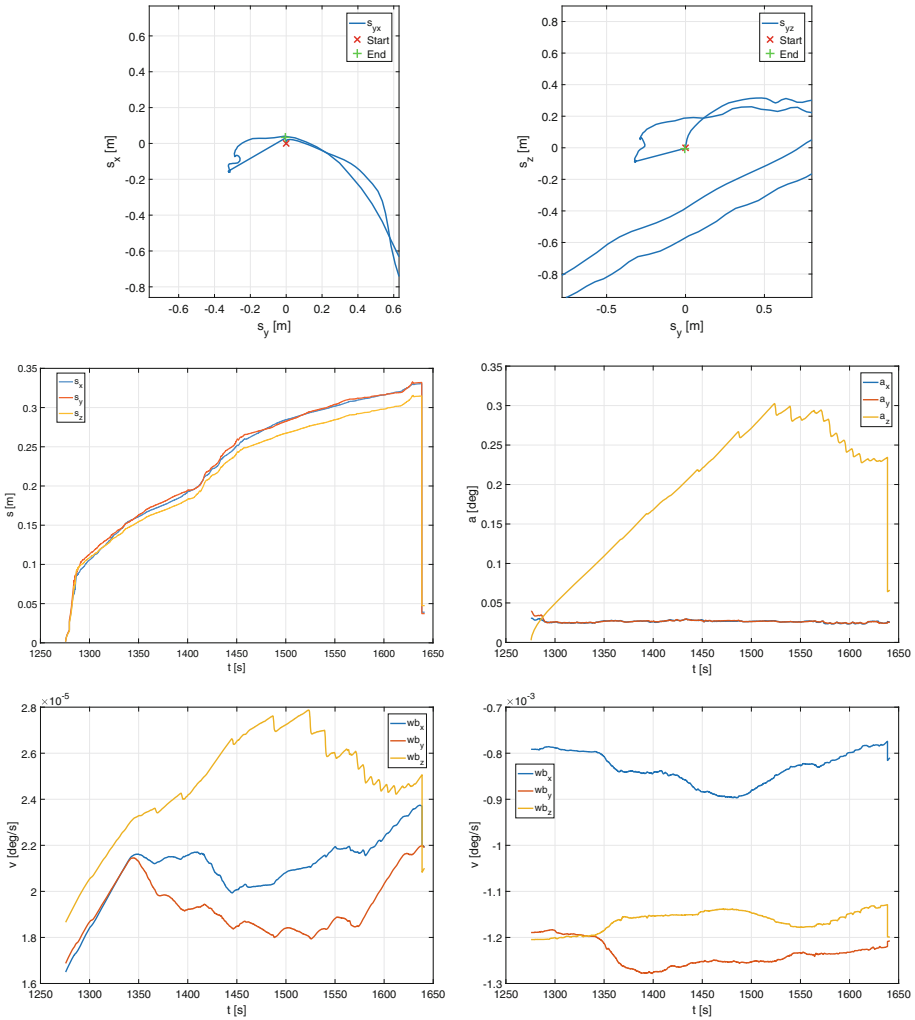


Fig. 6. *Top:* Both figures show the part of the measured trajectory. *Middle-left:* Uncertainties of measured positions. *Middle-right:* Uncertainties of measured rotation. *Bottom-left:* Uncertainties of modeled angular velocity biases. *Bottom-right:* Modeled angular velocity biases.

The start and end position of the dynamic route is the same point. Therefore, a loop-closure should be detected by the system at the end of the image sequence.

Figure 6 shows the effect of loop-closure. The trajectory is corrected by loop-closure measurements and jumps to a position very close to the start point.

Furthermore, all of the filter states (uncertainties, sensor biases) are updated at the same time. In this case, the sensor model is updated by the proposed method.

5 Conclusion

In this paper, an extended stochastic cloning method is proposed to overcome one of the toughest problems in the Kalman filter, the information-fusion problem of relative measurements which cross many periods of filter cycles. The testing results show the performance of the proposed method. The trajectory error is reduced to a very small number by loop-closure detection. Moreover, the biases of the sensors and the uncertainties of the system are reduced by involving keyframes and loop-closure detection, which means that the sensor model is updated by the proposed method. In our future work, we will explore how the updated sensor model affects the whole positioning system.

References

1. Griebbach, D., Baumbach, D., Zuev, S.: Stereo-vision-aided inertial navigation for unknown indoor and outdoor environments. In: IPIN (2014)
2. Grisetti, G., Kümmerle, R., Stachniss, C., Burgard, W.: A tutorial on graph-based SLAM. *IEEE Intell. Transp. Syst. Mag.* **2**(4), 31–43 (2010)
3. Brink, W., Van Daalen, C., Brink, W.: FastSLAM with stereo vision. In: 23rd Annual Symposium of the Pattern Recognition Association of South Africa (2012)
4. Gui, J., Gu, D., Wang, S., Hu, H.: A review of visual inertial odometry from filtering and optimisation perspectives. *Adv. Robot.* **29**(20), 1289–1301 (2015)
5. Strasdat, H., Montiel, J.M.M., Davison, A.J.: Real-time monocular SLAM: why filter? In: IEEE International Conference on Robotics and Automation (2010)
6. Strasdat, H., Montiel, J.M.M., Davison, A.J.: Visual SLAM: why filter? *Image Vis. Comput.* **30**(2), 65–77 (2012)
7. Mur-Artal, R., Tardos, J.D.: ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras. *IEEE Trans. Robot.* **33**(5), 1255–1262 (2017)
8. Younes, G., Asmar, D., Shammas, E., Zelek, J.: Keyframe-based monocular SLAM: design, survey, and future directions. *Robot. Auton. Syst.* **98**, 67–88 (2017)
9. Williams, B., Cummins, M., Neira, J., Newman, P., Reid, I., Tardos, J.: A comparison of loop closing techniques in monocular SLAM. *Robot. Auton. Syst.* **57**, 1188–1197 (2016)
10. Roumeliotis, S.I., Burdick, J.W.: Stochastic cloning: a generalized framework for processing relative state measurements. In: IEEE International Conference on Robotics and Autonomous (2002)

Author Index

- Ancé, Sarah 160
Aoki, Yoshimitsu 127
- Bach, Viet-Dung 110
Baumbach, Dirk 263
Berg, Maxime 150
Besier, Thor 223
Börner, Anko 263
Bourgais, Aurélien 160
- Cao, Dong 12
Chang, Yunpeng 187
Chen, Shizeng 3
Chung, Chung Choo 20
- Dinh, Sang Viet 196
Doan, Huong-Giang 110, 196
- Finch, Mark 223
Fischer, Christian 235
- Grießbach, Denis 263
Guo, Yuan 3
- Halle, Winfried 235
Han, Fei 211
Haneishi, Hideaki 141
Ho, Harvey 150, 160
Hohn, Rüdiger 253
Hosoya, Kento 169
Huang, Weilun 211
- Imiya, Atsushi 169
- Ji, Wanting 74
Jiang, Hao 47
- Kato, Hideyuki 141
Katsura, Koyo 37
Kawabe, Morio 141
Kawashima, Koki 37
Kempa-Liehr, A. W. 223
- Kim, Seunghyun 20
Kim, Whoi-Yul 20
Klette, Reinhard 29
Kokura, Yuri 141
- Le, Thi-Lan 196
Lee, Dabeen 20
Lee, Hongjun 20
Lee, Joo-Ho 100
Lei, Junfeng 29
Li, Bijun 3
Li, Zeya 74
Litifu, Ayixiamu 47
Liu, Xiaoxu 61
Liu, Zhenbing 74
Luo, Bin 187
- Nakano, Kazuya 141
Neuyen, Minh 61
Nguyen, Duy-Cuong 196
Nguyen, Thi-Oanh 196
Nguyen, Thi-Thuy 196
Nguyen, Van-Toi 196
Nozawa, Kouki 169
- Obikane, Shun 127
Ohnishi, Takashi 141
Ooka, Yoshihiko 141
Oram, Jonty 223
- Qin, Qianqing 187
Qiu, Zirui 211
- Reulke, Ralf 235, 253
- Sakai, Tomoya 141
Skegg, Michael P. 253
- Terzibaschian, Thomas 235
Tian, Yan 74
Tran, Thanh-Hai 110, 196
Tu, Zhigang 187

Vu, Hai 196

Wang, Jihua 47

Wang, Ruili 74

Wong, Andrew 223

Wu, Longyong 211

Wu, Yiqi 211

Wu, Yuchen 85

Xiao, Jinsheng 29, 47

Xie, Honggang 29

Xie, Wenjuan 29

Xu, Lisha 12

Yamane, Sasuke 100

Yamazoe, Hirotake 100

Yan, Wei Qi 61

Yan, Yuchen 47

Yanai, Keiji 85

Yao, Weiqing 47

Zell, Adina 235

Zhang, Changwei 150

Zhang, Dejun 211

Zhang, Hongmou 263

Zhang, Runtong 85

Zhou, Jian 3

Zong, Ming 74