Pier Luigi Mazzeo
Paolo Spagnolo
Thomas B. Moeslund (Eds.)

Activity Monitoring by Multiple Distributed Sensing

LNCS 8703

Second International Workshop, AMMDS 2014
Stockholm, Sweden, August 24, 2014
Revised Selected Papers

Springer

Lecture Notes in Computer Science 8703

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

Editorial Board

More information about this series at http://www.springer.com/series/7409

Pier Luigi Mazzeo · Paolo Spagnolo
Thomas B. Moeslund (Eds.)

Activity Monitoring by Multiple Distributed Sensing

Second International Workshop, AMMDS 2014
Stockholm, Sweden, August 24, 2014
Revised Selected Papers

Springer

Editors
Pier Luigi Mazzeo
Istituto Nazionale di Ottica - CNR
Lecce
Italy

Thomas B. Moeslund
Aalborg University
Aalborg
Denmark

Paolo Spagnolo
Istituto Nazioinale di Ottica - CNR
Lecce
Italy

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-319-13322-5 ISBN 978-3-319-13323-2 (eBook)
DOI 10.1007/978-3-319-13323-2

Library of Congress Control Number: 2014956227

LNCS Sublibrary SL3 – Information Systems and Applications, incl. Internet/Web and HCI

Springer Cham Heidelberg New York Dordrecht London

Printed on acid-free paper

Springer International Publishing AG Switzerland is part of Springer Science+Business Media (www.springer.com)

Preface

Nowadays different scientific research communities have oriented their efforts toward intelligent recognition of activity in distributed sensing environment.

An increasing number of algorithms and applications use a huge amount of different types of sensors, due to their relatively low cost and commercial diffusion.

A distributed sensor network includes a set of spatially scattered intelligent sensors designed to obtain measurements from the environment, to extract relevant information from the data gathered, and to infer appropriate decision from the information gained.

Distributed sensors network dimension depends on the multiple processors to simultaneously gather and process information from many different sources.

New technology availability makes these sensing networks economically feasible. The scope of this book is to investigate the problem of using distributed sensor networks to track, monitor, and understand the activity of human beings. This research field has different application areas such as human–computer interaction, user interface design, robot learning, and surveillance. At the highest decision level, the activity monitoring task addresses human behavior recognizing and intention understanding, from different observation sources. This is a very difficult task, even for humans to perform, where misinterpretations are common. This book collects different works presented at 2014 AMMDS Workshop in Stockholm. All chapters are centered on the application of distributed sensing network in the areas of human motion detection and tracking; human activity recognition; surveillance and security.

August 2014

Pier Luigi Mazzeo
Paolo Spagnolo
Thomas B. Moeslund

Organization

Committees

Technical Program Committee

Annalisa Milella	ISSIA-CNR, Italy
Marco Leo	Italian National Research Council, Italy
Simone Calderara	Università di Modena e Reggio Emilia, Italy
Andrea Prati	Università IAUV Venezia, Italy
Christian Micheloni	Università di Udine, Italy
George Bebis	University of Nevada, USA
Liliana Lo Presti	Università di Palermo, Italy
Guan Luo	Chinese Academy of Science, China
Federico Pernici	Università di Firenze, Italy
Wei-Shi Zheng	Sun Yat-sen University, China
Donato Di Paola	ISSIA CNR, Italy
Massimo Caccia	ISSIA CNR, Italy
Rama Chellappa	University of Maryland, USA
Sergio Escalera	University of Barcelona, Spain
Juergen Gall	Max Planck Institute, Germany
Shaogang Gong	Queen Mary University of London, UK
Jordi Gonzales	UAB-CVC, Catalonia, Spain
Amy Loutfi	Örebro University, Sweden
Cosimo Distante	Italian National Research Council, Italy
Sigal Leonid	Disney Reseach, USA
Richard Bowden	University of Surrey, UK
David Geronimo	KTH, Sweden
Bir Bhanu	University of California, Riverside, USA
Ugur Murat Erdem	University of Boston, USA

Contents

A Distributed Cooperative Architecture for Robotic Networks with Application to Ambient Intelligence

Antonio Petitti[1(✉)], Donato Di Paola[1], Annalisa Milella[1], Pier Luigi Mazzeo[2], Paolo Spagnolo[2], Grazia Cicirelli[1], and Giovanni Attolico[1]

[1] Institute of Intelligent Systems for Automation (ISSIA), National Research Council (CNR), via G. Amendola, 122 D/O, 70126 Bari, Italy
{petitti,dipaola,milella,grace,attolico}@ba.issia.cnr.it
[2] National Institute of Optics (INO), National Research Council (CNR), via Barsanti SNC, 73010 Arnesano, LE, Italy
{pierluigi.mazzeo,paolo.spagnolo}@ino.it

Abstract. A Distributed Cooperative Architecture (DCA) for applications of Ambient Intelligence is presented. The proposed cooperative system is composed by several static cameras and by a team of multi-sensor mobile robots. The nodes of the robotic network can act with some degree of autonomy and can cooperate to perform general purpose complex tasks such as distributed people tracking. The paper describes the system architecture and illustrates the feasibility of the proposed approach through preliminary experimental results.

1 Introduction

The design and development of systems for Ambient Intelligence (AmI) is an active investigation field, with many potential applications, including safety, security, and health-care assistance [8,13]. One of the main research challenges in the AmI domain is activity monitoring, which, in turn, depends on accurate and robust people tracking. This can be achieved by means of distributed sensor networks. As a basic principle, in distributed estimation, each node of the network locally estimates the state of a dynamical process using information provided by its local sensor and by a subset of nodes of the network, called neighbors [17]. In the literature, several approaches to distributed estimation in sensor networks can be found. Their particular characteristic is the presence of an agreement step that aims at minimizing the discrepancy among sensory nodes [2,3,6]. Among the various sensors, cameras have been especially investigated as an effective solution for environmental monitoring. In [14], data fusion and tracking methods for decentralized and distributed camera networks are discussed. A review of distributed algorithms for several computer vision applications can also be found in [15], emphasizing the advantages of distributed approaches with respect to centralized ones.

The use of multiple sensors increases reliability and effectiveness in large environments. As a drawback, it imposes the need of modifying infrastructures that

P.L. Mazzeo et al. (Eds.): AMMDS 2014, pp. 1–12, 2014.
DOI: 10.1007/978-3-319-13323-2_1

Fig. 1. Conceptual representation of the proposed architecture for ambient intelligence.

can be heavy and expensive. A smart and innovative solution to this problem is the exploitation of flexible moving sensors that can be mounted on semi or fully autonomous vehicles. These vehicles represent mobile nodes of the distributed network. They can be employed as individual agents or organized in teams to provide intelligent distributed monitoring of broad areas. Mobile sensors may significantly expand the potential of AmI technologies beyond the traditional passive role of event detection and alarm triggering from a static point of view. Mobile robots can actively interact with the environment, with humans or with other robots to accomplish more complex cooperative actions [1,7,16]. Nevertheless, mobile surveillance devices based on autonomous vehicles are still in their initial stage of development and many issues are currently under investigation [4,5,10].

In this paper, a Distributed Cooperative Architecture (DCA) is presented. It integrates fixed and mobile heterogeneous sensors to intelligently monitor large environments and track people. Figure 1 shows a conceptual representation of the system, which includes fixed calibrated cameras and a team of autonomous mobile robots equipped with different sensors. The system is being developed as part of the Italian National Research Program PON-BAITAH - "Methodology and Instruments of Building Automation and Information Technology for

pervasive models of treatment and Aids for domestic Healthcare", which is aimed to develop ICT AmI technologies to support fragile people in their domestic environments.

In the proposed system, mobile sensors supply two main functionalities: (1) they can supply information about the observed human target in areas not surveyed by the fixed cameras; (2) they can move close to the target to increase precision and reliability of scene analysis whenever fixed sensors are unable to provide robust estimates.

The main contribution of this work is related to the design of the DCA, which involves different challenges. The major one is the integration of high-level decision-making issues with primitive simple behaviors for different operative scenarios. This aim requires a modular and reconfigurable system, capable of simultaneously addressing low-level reactive control, general purpose and monitoring tasks and high-level control algorithms in a distributed fashion. The details on the DCA and its main components are described in Sect. 2. Preliminary real-world experiments using the proposed DCA system are presented in Sect. 3. Finally, conclusions are drawn in Sect. 4.

2 Distributed Cooperative Architecture

The Distributed Cooperative Architecture (DCA) is a control architecture for heterogeneous networks of sensors and robots, hereinafter referred to as agents. All the agents form a peer-to-peer network, and differ for their own sensor capabilities. Specifically, every agent is able to detect an event (e.g., to perceive moving people or objects) and to localize an event (e.g., tracking the position of a person) in the environment using one or more sensor devices; in addition, mobile agents are able to execute tasks, through their actuators. Both the static cameras and the mobile agents take advantage of people detection modules, which provide the input to a distributed target tracking algorithm, namely the Consensus-Based Distributed Target Tracking (CDTT) algorithm, previously proposed by some of the authors in [12]. The CDTT module constitutes the main layer of the DCA. In the following, first, the CDTT algorithm, is recalled. Then, a detailed description of fixed and mobile agents is provided.

2.1 Distributed Target Tracking Algorithm

The core functionality of the DCA system is the tracking of people, based on the *fully distributed* Consensus-based Distributed Target Tracking (CDTT) algorithm. This algorithm is aimed to enhance the performance of the people tracking task in a heterogeneous sensor network. It consists of a two-phase iterative procedure: an estimation step and a consensus step. In the estimation phase, each node of the network gives an estimate of the position of the target. If the node can directly take a measurement, then it will estimate the target position by means of a Kalman filter. Otherwise, the node will take a prediction of the target motion according to the embedded linear motion model of the Kalman filter.

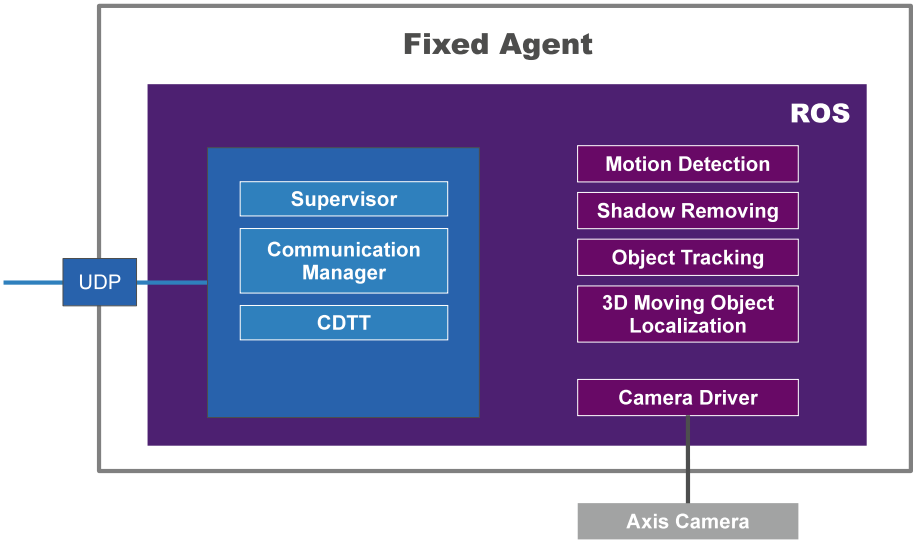


Fig. 2. Schematic representation of interconnections among modules composing the Fixed Agent node.

In the consensus phase, all the estimates in the network converge to a common value via a *max-consensus* protocol, performed on a measurement accuracy metrics called *perception confidence value*. This approach was proved to provide good performance in heterogeneous sensor networks composed by nodes with limited sensing capabilities [6]. The CDTT approach is totally distributed, as it does not involve any form of centralization. Moreover, it guarantees the agreement of the network nodes on the target position. The reader is referred to [12] for further information.

2.2 Fixed Agents

The Fixed Agents run on a workstation linked to each camera by a network infrastructure. The schematic representation of interconnections among the components of the Fixed Agent module is shown in Fig. 2. The Fixed Agents are implemented in the Robot Operating System (ROS)[1] framework. In particular, each Fixed Agent is a ROS node in which two main components run: the *perception* module and distributed *target tracking* module.

The perception module includes a set of functionalities illustrated in Fig. 2 (i.e., motion detection, shadow removal, object tracking, 3D moving object localization) aimed to robustly detect and localize people with respect to the cameras. Details about the developed algorithms can be found in [11].

[1] http://www.ros.org

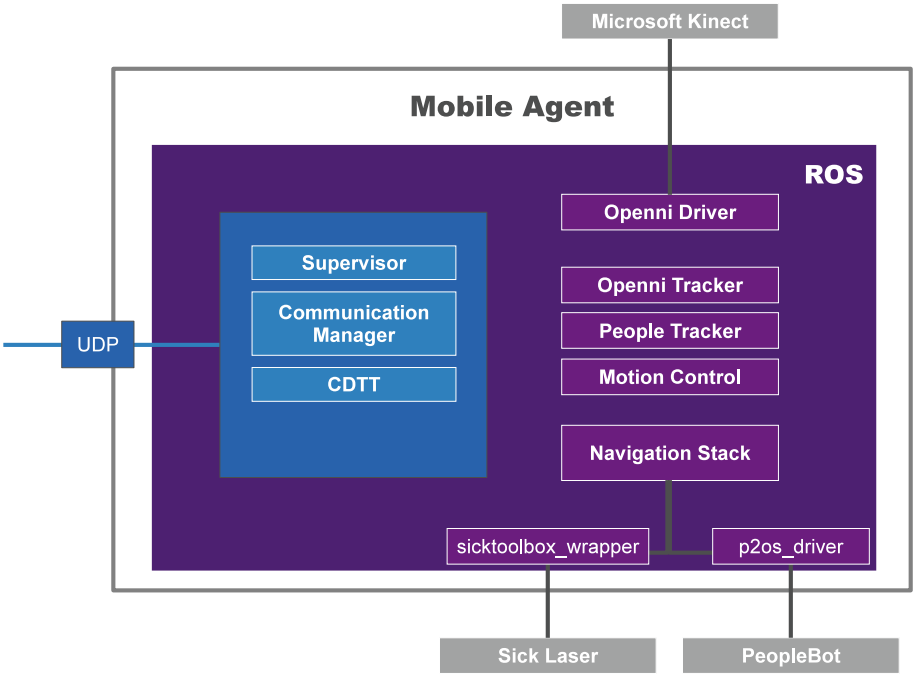


Fig. 3. Schematic representation of interconnections among modules composing the Mobile Agent node.

The distributed target tracking module implements the CDTT algorithm, described in the previous subsection, and the communication procedures, which allow the cooperation of agents via the UDP protocol. The UDP protocol is used for the inter-agent communication because, with the multicast technique, it avoids the need for a centralized hub.

2.3 Mobile Agents

The Mobile Agents of the network consist of mobile robots. Each mobile agent is equipped with sensory devices to interact with the environment. Every mobile agent is able to localize itself in the environment and to safely navigate avoiding static and dynamic obstacles. It is also able to identify and track the position of a target in the environment. ROS has been adopted as a framework for communication management, sensor acquisition and actuator control. ROS provides a *Navigation Stack*, which enables the robot to navigate in a known environment avoiding obstacles, as well as sensor management packages [9]. The structure of the navigation stack of ROS was modified so as to develop a customized monitoring architecture enriched with new functionalities with respect to the native ROS framework. Specifically, surveillance capabilities were added to the mobile nodes. A coordinate transformation from local to global coordinates was also

introduced for the people tracking task. A schematic representation of a mobile agent is shown in Fig. 3. All ROS nodes run on the on-board laptop, except for *sicktoolbox_wrapper* and *p2os_driver*, which run on the embedded pc of the robot. As can be seen, the Navigation Stack of ROS produces robot position estimates, as well as information about obstacles on the basis of laser measurements. The ROS node *motion_control*, implemented by our research team, sends velocity references to *p2os_driver* ROS node, responsible of the robot guidance. The *people_tracker* node estimates the relative position of people with respect to the robot. It is based on the *openni_tracker*, which uses input from an onboard Kinect camera. The relative coordinates of detected people, transformed in the world reference frame, provide input data to the distributed target tracking algorithm.

3 Experimental Results

In [11], the proposed system was validated through numerical simulation campaigns, showing that the presence of a mobile node in addition to the fixed agents improves the tracking accuracy. Here, experimental tests conducted in a real-world scenario are presented. First, the experimental setup is described, then results of the experimental tests are reported.

3.1 Environment Setup

The experimental setup is shown in Fig. 4. The picture displays the map of a corridor of the ISSIA-CNR building, as obtained by the ROS *gmapping* node using laser data acquired by a mobile robot during a complete exploration of the environment. In this experimentation, three fixed cameras and one mobile robot were employed. The positions of the fixed cameras (C_1, C_2, C_3) and of the initial position of the mobile robot (R_1) are overlaid on the map. Using its onboard sensors, the mobile agent is able to localize itself in the environment and to carry out surveillance tasks, such as people detection and tracking. Cameras are calibrated, therefore events detected in the image plane can be located in the real world and their positions can be communicated to the mobile agent. The mobile robot can explore areas that are unobservable by the fixed cameras, thus improving the accuracy in detecting events by reaching proper positions in the environment. Hence, the proposed system could be useful to reduce the number of fixed sensors or to monitor areas (e.g., cluttered environments) in which the field of view of the fixed cameras can be temporarily and dynamically reduced.

3.2 Sensor Network Setup

The fixed nodes consist of three wireless IP cameras (C_1, C_2, C_3) located in different places of the environment (see map in Fig. 4). C_2 and C_3 are Axis IP color cameras with a 640×480 pixel resolution and an acquisition frame rate of 10 frames per second. C_1 is a Mpixel Axis IP color camera with 1280×1024 pixel

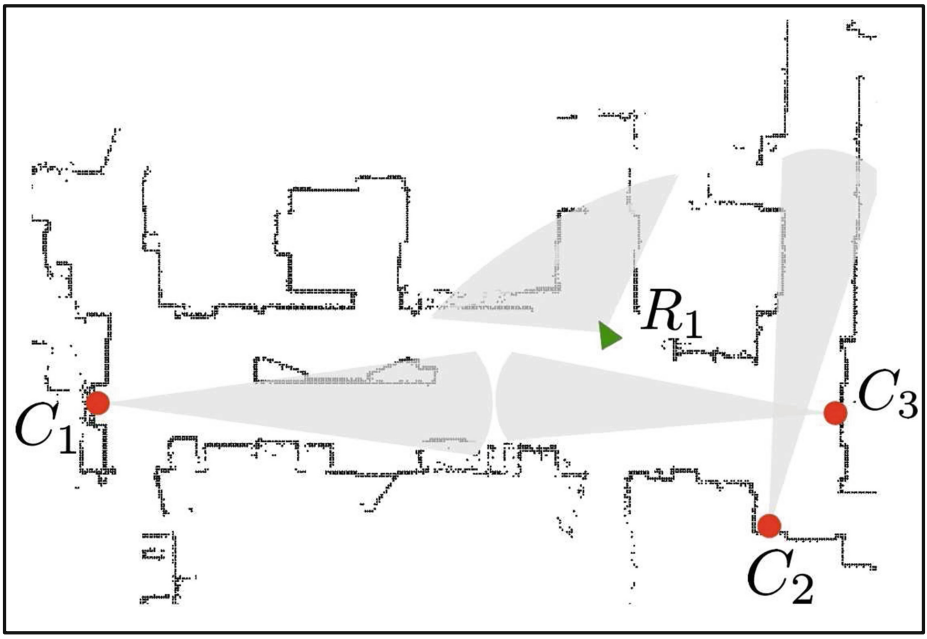


Fig. 4. Map of one corridor of the office with overlaid the position of three static cameras (red circles) and one mobile agent (green triangle) (Color figure online).

Table 1. Average MSE and variance in the tracking of a person moving in the laboratory by means of a network of 4 nodes, 3 fixed and 1 mobile.

Case	Average MSE [m]	Variance [m^2]
Trajectory 1 (Fig. 6)	1.15	0.86
Trajectory 2 (Fig. 7)	0.75	0.16

resolution and full frame acquisition rate of 8 frames per second (see Fig. 5, on the right). A calibration step to estimate intrinsic and extrinsic parameters was performed for each camera using the Matlab Calibration Toolbox[2]. This allows camera coordinates to be mapped to the map reference frame.

The mobile agent (denoted as R_1 in Fig. 4) is a PeopleBot mobile robot platform equipped with a laser range-finder, a Kinect, and an on-board laptop (see Fig. 5, on the left). The SICK laser is connected with the embedded robot control unit. The Kinect camera and the PeopleBot control unit are connected with the laptop, via a USB cable and a crossover cable, respectively. The laser range-finder is used to build a map of the environment and to localize the vehicle.

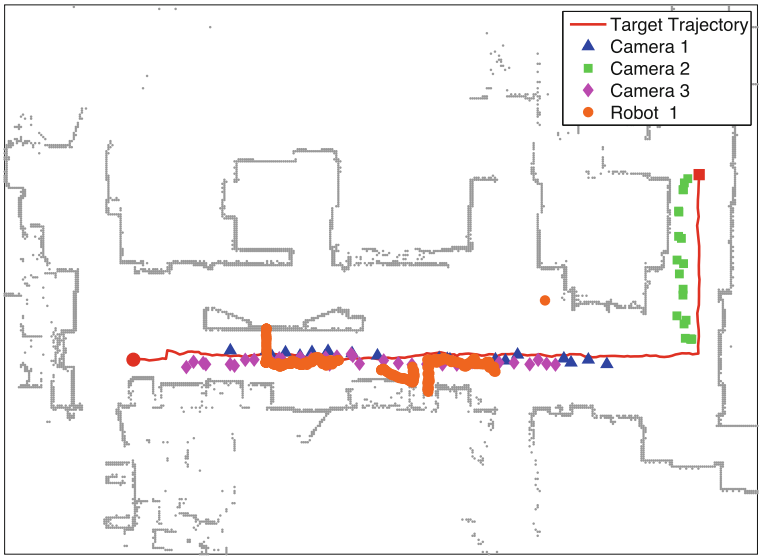
[2] The toolbox is available on http://www.vision.caltech.edu/bouguetj/calib_doc/index.html.

Fig. 5. The nodes of the network. On the left, the mobile agent PeopleBot. The robot is equipped with a laser range-finder SICK LMS200 and a Kinect. On the right, two different AXIS cameras: on the top, a Mpixel Axis IP color camera with 1280×1024. On the bottom, an Axis IP color cameras with a 640×480 pixel camera.

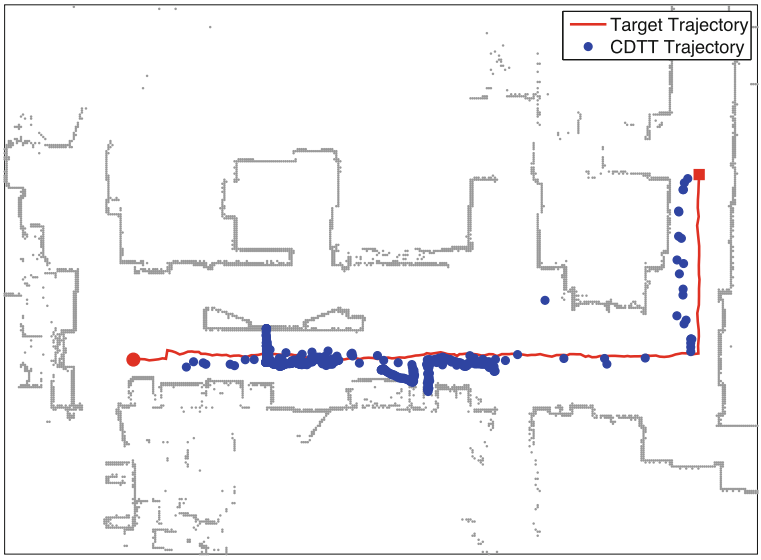
The Kinect is used for both navigation (e.g., obstacle avoidance) and high-level tasks, such as people detection and tracking.

3.3 Results of Experiments

A network of three fixed cameras and one robot, realized in our lab, was used to verify the DCA system performance in a real application: to track a person moving in the environment. The two different trajectories followed by the target during the experimentation are shown in Figs. 6 and 7. In both Fig. 6(a) and Fig. 7(a), the red line represents the real trajectory of the target, while different markers represent the sequence of target positions estimated by each single node and by the monitoring network as a whole using the DCA architecture. In particular, Fig. 6(b) and Fig. 7(b) compare the target trajectory (red line) with the trajectory estimated by the CDTT algorithm (blue dots). The CDTT algorithm requires all the network nodes to share the same information about the target

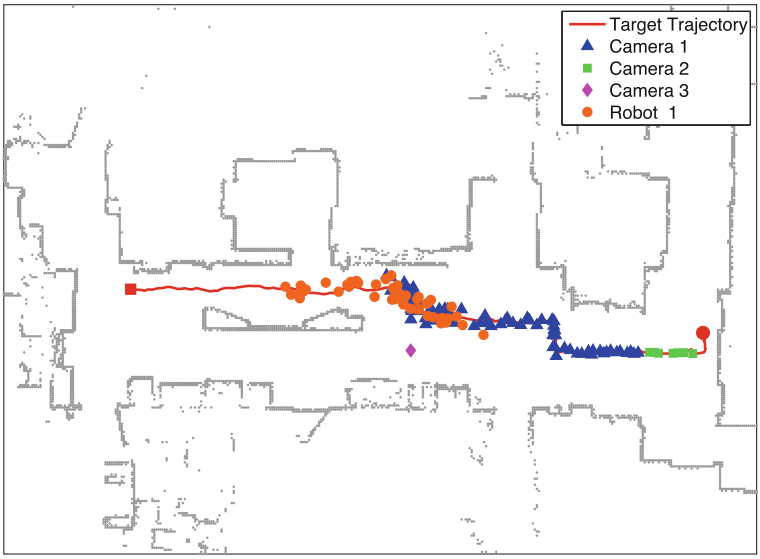


(a)

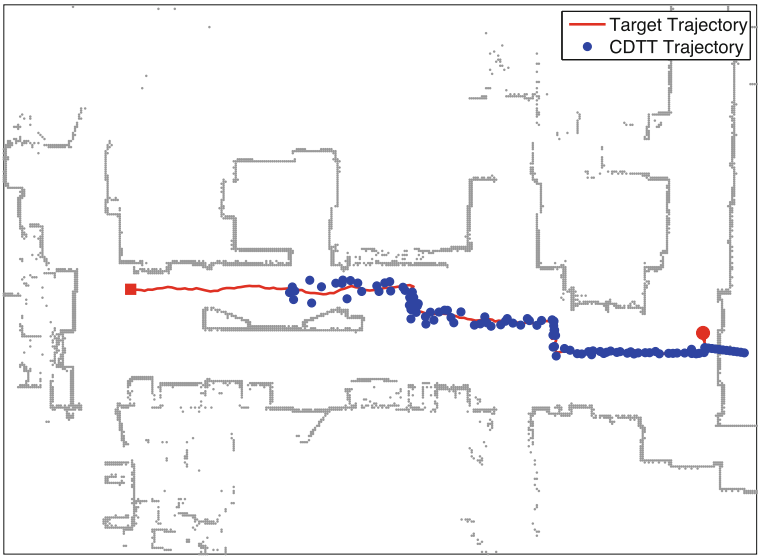


(b)

Fig. 6. Trajectory 1. The measurement of the position of the target carried out by each sensor of the network (a) and the CDTT trajectory recovered on line and in distributed fashion by the network (b) (Color figure online).



(a)



(b)

Fig. 7. Trajectory 2. The measurement of the position of the target carried out by each sensor of the network (a) and the CDTT trajectory recovered on line and in distributed fashion by the network (b) (Color figure online).

location after convergence of the consensus step. Therefore, the estimated target position is the same for any node of the network. To quantify the tracking performance, the target is supposed to move at constant velocity to allow the calculation of the MSE. The collected results, shown in Table 1, exhibit a mean square error of 1.15 m and 0.75 m, for Trajectory 1 and Trajectory 2, respectively.

4 Conclusions

The paper introduced a distributed cooperative architecture for applications of ambient intelligence. Its main contribution is a monitoring network composed by fixed and mobile nodes. The use of mobile nodes produces a twofold advantage: it allows the complete coverage of large environments with a lower number of sensors (with respect to the use of fixed nodes), and it increases the accuracy of measurements by deploying the sensors in the most favorable positions to observe the current target. The global control architecture used by the system was presented and the software agents developed for both fixed and mobile nodes were described. The feasibility and effectiveness of the proposed system are shown by preliminary experimental results obtained using a monitoring network realized in our lab environment.

Acknowledgements. This research was supported by the National Operational Program (PON) for Research and Competitiveness 2007–2013, project BAITAH "methodology and instruments of Building Automation and Information Technology for pervasive model of treatment and Aids for domestic Health-care", code PON01_00980.

The authors thank Arturo Argentieri for technical support in the setup of the system presented in this work.

References

1. Burgard, W., Moors, M., Fox, D., Simmons, R., Thrun, S.: Collaborative multi-robot exploration. In: Proceedings of IEEE International Conference on Robotics and Automation. ICRA '00, vol. 1, pp. 476–481 (2000)
2. Carli, R., Chiuso, A., Schenato, L., Zampieri, S.: Distributed Kalman filtering based on consensus strategies. IEEE J. Sel. Areas Commun. **26**(4), 622–633 (2008)
3. Cattivelli, F., Sayed, A.: Diffusion strategies for distributed Kalman filtering and smoothing. IEEE Trans. Autom. Control **55**(9), 2069–2084 (2010)
4. Di Paola, D., Milella, A., Cicirelli, G., Distante, A.: An autonomous mobile robotic system for surveillance of indoor environments. Int. J. Adv. Rob. Syst. **7**(1), 19–26 (2010)
5. Di Paola, D., Naso, D., Milella, A., Cicirelli, G., Distante, A.: Multi-sensor surveillance of indoor environments by an autonomous mobile robot. In: IEEE 15th International Conference on Mechatronics and Machine Vision in Practice, pp. 23–28 (2008)
6. Di Paola, D., Petitti, A., Rizzo, A.: Distributed Kalman filtering via node selection in heterogeneous sensor networks. Int. J. Syst. Sci., 1–12 (2014). http://dx.doi.org/10.1080/00207721.2013.873836

7. Giannini, S., Di Paola, D., Rizzo, A.: Coverage-aware distributed target tracking for mobile sensor networks. In: 2012 IEEE 51st Annual Conference on Decision and Control (CDC), pp. 1386–1391, Dec 2012
8. Magherini, T., Fantechi, A., Nugent, C., Vicario, E.: Using temporal logic and model checking in automated recognition of human activities for ambient-assisted living. IEEE Trans. Human-Mach. Syst. **43**(6), 509–521 (2013)
9. Marder-Eppstein, E., Berger, E., Foote, T., Gerkey, B., Konolige, K.: The office marathon: robust navigation in an indoor office environment. In: International Conference on Robotics and Automation (2010)
10. Milella, A., Di Paola, D., Mazzeo, P., Spagnolo, P., Leo, M., Cicirelli, G., D'Orazio, T.: Active surveillance of dynamic environments using a multi-agent system. In: 7th IFAC Symposium on Intelligent Autonomous Vehicles. IAV 2010, vol. 7, pp. 13–18 (2010)
11. Petitti, A., Di Paola, D., Milella, A., Mazzeo, P., Spagnolo, P., Cicirelli, G., Attolico, G.: A distributed heterogeneous sensor network for tracking and monitoring. In: 2013 10th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 426–431, Aug 2013
12. Petitti, A., Di Paola, D., Rizzo, A., Cicirelli, G.: Consensus-based distributed estimation for target tracking in heterogeneous sensor networks. In: 2011 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC), pp. 6648–6653 (2011)
13. Rashidi, P., Mihailidis, A.: A survey on ambient-assisted living tools for older adults. IEEE J. Biomed. Health Inf. **17**(3), 579–590 (2013)
14. Taj, M., Cavallaro, A.: Distributed and decentralized multicamera tracking. IEEE Signal Process. Mag. **28**, 46–58 (2011)
15. Tron, R., Vidal, R.: Distributed computer vision algorithms. IEEE Signal Process. Mag. **28**, 32–45 (2011)
16. Vig, L., Adams, J.: Coalition formation: from software agents to robots. J. Intell. Rob. Syst. **1**, 85–118 (2007)
17. Xu, Y., Gupta, V., Fischione, C.: Distributed estimation. In: Chellappa, R., Theodoridis, S. (eds.) E-Reference Signal Processing. Elsevier, Oxford (2013). http://www.ee.kth.se/~carlofi/Publications/e-reference-distributed-estimation.pdf

A Customizable Approach for Monitoring Activities of Elderly Users in Their Homes

Jonas Ullberg(✉), Amy Loutfi, and Federico Pecora

Center for Applied Autonomous Sensor Systems,
Örebro University, 70182 Örebro, Sweden
{jonas.ullberg,amy.loutfi,federico.pecora}@oru.se
http://www.oru.se/aass/

Abstract. This paper presents an implemented context recognition system that enables caregivers to query and visualize daily activities of elderly who live in their own homes. The system currently serves several homes across Europe and provides caregivers with the ability to correlate activities with specific health indicators. The system also allows to define conditions under which alarms should be raised.

1 Introduction

Encouraging independent living as a way to promote a healthier society is a social and economic challenge. Elderly people wish to remain in their homes as long as possible as this gives them a richer social life and helps them maintain established habits. Enabling old people to do so is also positive from an economic perspective as the cost of home care is less than the cost of residential care. However, several issues need to be addressed in order to prolong independent living. An essential aspect among them is the early detection of possible deterioration of health so that problems can be remedied in an early stage and with the timely involvement of health care professionals and family.

A key enabler in this respect is automated behavior monitoring over time. Monitoring solutions must possess two key qualities: (**requirement 1**) the ability to selectively focus on different aspects of daily life depending on circumstances that are assessed by a physician or family member; and (**requirement 2**) the ability to trace these aspects over medium to long periods of time. To mention a few, interesting health-affecting behaviors can be; decrease of physical activity, irregularity in sleep, changes in cooking and eating habits and so on.

This paper presents a context recognition system that addresses the two requirements above. The system infers and records the activities and status of elderly over extended periods of time. The specific way in which behaviors are recognized are specified through *temporal models*, which can be defined, added or removed dynamically to a list of behaviors of interest. Caregivers, medical experts and family members (henceforth, *secondary users*) can search

Jonas Ullberg: This article has been written under GiraffPlus EU grant (Contract no. 288173).

P.L. Mazzeo et al. (Eds.): AMMDS 2014, pp. 13–25, 2014.
DOI: 10.1007/978-3-319-13323-2_2

and inspect the recorded information through a versatile user interface which supports real time viewing of what is happening in the elderly person's home. The interface also aggregates and provides tools to analyze data extending over long periods of time. The models used for behavior tracking are specified in the form of qualitative relations among sensor readings that are definable by secondary users.

The context recognition is part of a larger system called GiraffPlus [4] and is developed in an EU-FP7 funded project. The GiraffPlus system includes a network of sensors placed in the home or worn by the elderly. These include physiological sensors such as weight, blood pressure and pulse oxymetry, as well as environmental sensors like infrared motion, pressure, temperature, and electrical usage sensors. Data from these sensors are processed by the context recognition system through the qualitative models provided by secondary users. The GiraffPlus system is named after one of its components: the Giraff telepresence robot. The robot uses a Skype-like interface to allow caregivers to virtually visit an elderly person in the home. The Giraff telepresence robot is used as the primary means of communicating with the elderly; results of trend analysis, details of the models used for monitoring, and currently active sensors can all be accessed and/or manipulated on the Giraff robot's interface.

In addition to providing behavioral traces for secondary users, the context recognition infrastructure also synthesizes appropriate action plans to aid the primary user when certain conditions hold on the recognized behaviors. Such enactments manifest themselves as proactive alerts, e.g. if the user is recognized as having altered sleep patterns over several days, appropriate physiological and activity-related information is presented to the caregivers upon contact through the Giraff telepresence robot. Each secondary user can select contexts of interest by using their personal interface to the GiraffPlus system.

2 Background

Current approaches to the problem of recognizing human activities can be roughly categorized as *data-driven* or *model-driven*. In data-driven approaches, models of human behavior are acquired from large volumes of data over time. Notable examples of this approach employ Dynamic Bayesian network (DBNs) in conjunction with learning techniques for inferring transition probabilities [14,25]. Extensions of these approaches have been proposed for dealing with realistic features of the domain, such as interleaved activities [6,12] and multiple persons [21].

Although highly effective in specific domains, such systems are typically brittle to changes in the nature and quantity of sensors, requiring significant re-training when the application context changes. This contrasts with the requirement that the criteria for context recognition can be specified on-line depending on circumstances assessed by secondary users and put to service immediately, without the need for re-training and model tuning (**requirement 1**). Liao et al. [10] have described an approach which partially overcomes these limitations using conditional random fields, showing that learned behavior models can be generalized

to different users. However, this has been empirically proved only for the specific context of activity recognition using GPS traces and location information, and does not address the problem of contextual recognition and planning/execution. A complementary approach is followed by Helaoui et al. [8] to overcome some of the limitations of purely data-driven techniques. Specifically, the authors incorporate modeling capabilities to capture features such as qualitative temporal relations which describe how events relate to each other. One of the key features of the approach is its capability to recognize interleaved activities. However, it is limited to detecting sensor context, and the applicability of the approach to a highly-dynamic context, like our use case, is untested.

Model-driven approaches to activity recognition follow a complementary strategy in which patterns of observations are modeled from first principles rather than learned or inferred from large quantities of data. Such approaches typically employ an abductive process, whereby sensor data is explained by hypothesizing the occurrence of specific human activities[1]. Examples include work by Goultiaeva and Lespérance [7], where the Situation Calculus is used to specify very rich plans, as well as the work of Pinhanez and Bobick [16], Augusto and Nugent [2], Jakkula et al. [9], all of whom propose rich temporal representations to model the conditions under which patterns of human activities occur. Other techniques used to perform context recognition include ontological reasoning. For instance, Springer and Turhan [22] employ OWL-DL to specify models of complex situations, the argument being that the more complex the situation to recognize, the more sophisticated the behavior of the smart environment. However, time is considered only implicitly. Riboni and Bettini [19] combine ontological and statistical reasoning to reduce errors in context inference, albeit without addressing temporal relationships between activities.

Data- and model-driven approaches have complementary strengths and weaknesses: the former provide an effective way to recognize elementary activities from large amounts of continuous data – relying, however, on the availability of accurately annotated datasets for training; conversely, model-driven approaches provide a means to easily customize the system to different operational conditions and users through expressive modeling languages – which, though, is based on the ability of a domain modeler to identify criteria for recognition appropriately from first principles.

Constraint-based modeling and inference have been employed to perform schedule execution monitoring for domestic activities. Two notable representatives of this direction are described by Pollack et al. [17] and Cesta et al. [3]. These systems differ from our work in that they employ pre-compiled (albeit highly flexible) schedules as models for human behavior.

The context recognition engine used in the GiraffPlus system is mostly related to temporal constraint-based approaches such as SAM [15] and constraint-based chronicle recognition [5]. These approaches employ temporal reasoning techniques to perform on-line recognition of temporal patterns of sensory events. An approach based on evidence theory augmented with temporal features presented

[1] An approach similar to the work of [20] on deducing context in a robot's environment.

by Mckeever et al. [11] underscores the advantage of explicitly accounting for activity durations. Our work introduces a key novelty in temporal constraint-based context recognition, namely the ability to take temporal uncertainty in the sensor readings [24] into account. This capability is an important enabler of configurable (**requirement 1**) and continuous (**requirement 2**) recognition, as this allows us to interpret the output in time of sensors in ways that fit high-level, user-defined models of behavior, and possesses the necessary good performance to be used on-line.

In summary, GiraffPlus extends the state-of-the-art in context recognition in terms of (1) models of human behaviors that are instantiated on-line, (2) generalization of activity recognition to context recognition by taking multiple sources of physiological and environmental data into account, and (3) applicability to real world scenarios.

3 The GiraffPlus System

The GiraffPlus system integrates components for environmental sensing, physiological sensing, context recognition, data visualization and storage. In this section we briefly present the components of the system and how they are integrated in order to provide a better understanding of the role and place of the context recognition in the overall architecture. Figure 1 shows how the components of the system interacts with the services and hardware used in the project, the main components are as follows:

Physical Environment. The home of the end-user contains several sensors which are wirelessly connected to an Asus EEE Box PC which in turn is connected to Internet.

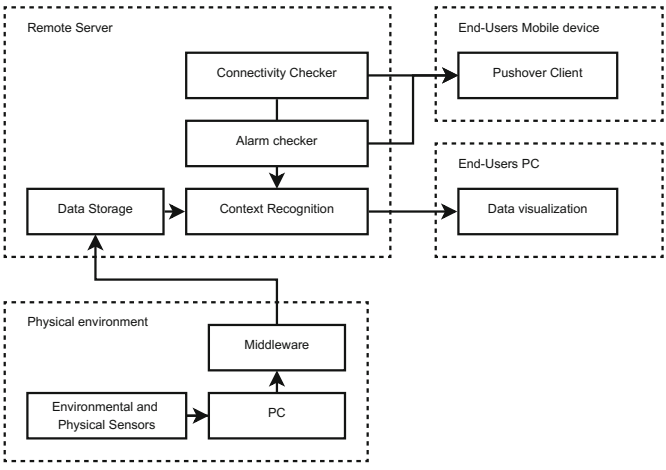


Fig. 1. A high level overview of the relationship between the context recognition and the other components in the GiraffPlus system.

Environmental and Physical sensors. The environmental sensors that are used include motion sensors, pressure sensors (to detect the presence of the inhabitant in the bed for instance), electrical usage sensors, reed-switch sensors (to detect open doors etc.), smoke alarms, flood detectors and more. All environmental sensors are provided by Tunstall, whereas the physiological sensors are provided by Intellicare. The latter measures physiological parameters such as blood pressure, heart rate and body weight. However, unless told otherwise, the end user takes these measurements whenever he or she so wishes, and the data these sensors provide are useful as-is to a caregiver and thus not deemed very interesting from a context recognition perspective. Therefore, the context recognition has focused on using the environmental sensors (although there is no limitation on this, both types of data are handled equally).

PC. The PC an Asus EEE Box PC, originally intended to be used as a media center, this computer is suitable to be used in homes due to its small form factor, low noise and power efficiency. The PC is connected to the Internet, either via a 3G router or directly depending on the available options at each test site. Furthermore, it is connected to a Tunstall Connect+ gateway which enables it to receive and forward data from the environmental sensors.

Middleware. The middleware that runs on the PC is partially derived from the PERSONA project [23] and handles forwarding of sensor data to the remote database server. It also handles buffering of data in case the Internet connection is temporarily lost or congested, thus data that can not be submitted are queued for later transmission. A detailed description of the middleware can be found in [13].

Remote Server. The remote server hosts a database and several Java-servlets running on a Tomcat web server. This means that all connections to the server are done by Representational State Transfer (REST) HTTP calls. Since the data that is stored on the server is sensitive, all connections are protected with SSL and each user and home needs a personal certificate to connect to the server.

Data Storage. The data storage provides an API to query and store information about homes in a MongoDB database. This includes the sensor samples that are sent from the homes and other data such as information about primary and secondary users and their access rights. MongoDB is a document database that focuses on scalability. Scalability was deemed an important feature since a useful and commercially viable system needs to be able to support thousands of homes.

Context Recognition. The context recognition system is deployed on the same server as the database in order to remove the overhead of transmitting raw samples over Internet and to facilitate updates. A client sends a query to the CR consisting of a rule document and a time period. The server in turn requests the required data from the DB and infers a corresponding timeline containing the activities that were queried for.

Alarm Checker. This system regularly queries the context recognition module for user-defined alarm conditions. If an alarm condition is detected the system will send a Pushover notification to alert relatives and caregivers.

Connectivity Checker. This systems monitors the connectivity of the test sites and alerts technical support if a given home has not provided any data for a long period of time (due to issues with the Internet connection, the local PC or the sensor system).

End User Devices. The end user, which can be an elderly a relative or a caregiver, can use the Giraff system with a PC or a smart mobile device.

Pushover Client. This software runs on Android and iOS devices and presents short messages to the user. There are different levels of urgency to these messages which controls if the receiver is alerted with a sound during night or not for instance.

Data Visualization. The PC in the home or at the caregivers office runs the data visualization and personalization software "DVPIS". It enables the user to fetch the elderly's physiological measurements (e.g. body weight blood pressure etc.) and perform activity queries.

4 The Context Recognition Service

The context recognition is a REST service that runs as a servlet on a central Tomcat server. Queries to this service are done with a lightweight API which is embedded in several services that run on the client computer or on the central server. All computations are done on the central server when querying an activity. This architecture has the advantage of;

- Reducing bandwidth (since it does not need to transfer raw samples across the network).
- Allowing for a more strict access control to sensor data.
- Enabling system updates without requiring changes to client software.

Figure 2 shows the details of the context recognition service. The main point of interest is the inference procedure that is divided into three distinct steps; preprocessing, inference and extraction. The responsibilities of these are as follows.

Preprocessing Module. On the server the client sends a query to the context recognition engine by providing it with an XML-document describing how sensor data and activities correlate. The preprocessing module's responsibility is to fetch samples from the database and use these samples to build a higher level representation of the events that takes place in the home. This is done by using an appropriate preprocessor for the data. For instance, a timeline that declares

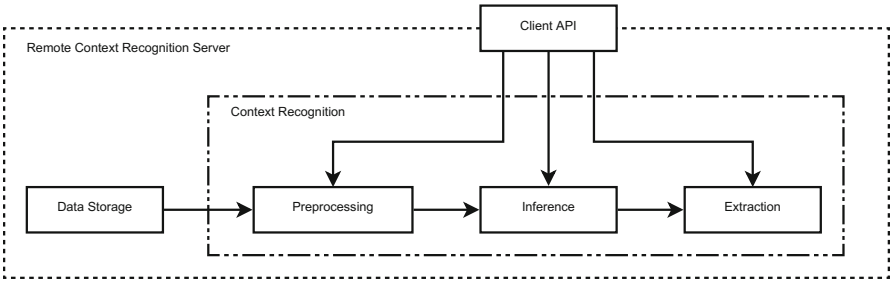


Fig. 2. This figure shows the flow of data within the context recognition module.

if a person is at a location or not, based upon PIR-motion sensors, can be constructed either by looking at individual sensors or by using sensors at other locations as well as terminal conditions. In the former case a temporal threshold parameter needs to be provided to determine the temporal extent to which a person is considered to be in a room, for this to work a continuous sequence of repeated motion readings needs to be generated by the user, and the query is parameterized with a maximum allowed temporal discontinuity between these. In the latter case the person is considered to be at a location until he is sensed somewhere else.

Inference Module. The symbolic models underlying the inference are grounded on a *constraint-based representation*. The key advantage of doing so lies in the widely recognized capability of this paradigm to support search and incremental constraint solving capabilities, and the relative efficiency of the resulting applications. The user-supplied rules used by the inference module define how sensor readings correlate to context that can be inferred. These correlations are expressed as temporal constraints in Allen's Interval Algebra [1] with metric bounds, however, the overall architecture supports the more expressive INDU algebra [18] which adds constraints on the relative duration of intervals. Activities are inferred by performing temporal constraint propagation on the domains of intervals generated by the preprocessing module and the output is a domain of intervals that are admissible with respect to the rules. The propagation and inference algorithm is described in detail in [24].

Extraction Module. The extraction module's responsibility is to generate timelines that can be used by other software components (e.g., the visualization software or the alarm system). As the inference and preprocessing module generates large amounts of hypotheses about the activities that have taken place there is the need to provide a system to easily analyze this data. In GiraffPlus this module only supports one type of extraction method, which extracts the maximum duration interval for an activity.

5 Evaluation in a Swedish Home

Since it is difficult to collect ground truth of performed activities (due to the fact that the elderly can't be asked to annotate what they are doing) an evaluation was done together with a local caregiver with insights into a test subject's daily life and medical history. The goal was to assess how well the system could infer medically meaningful information about the users daily life.

The apartment in this case study is inhabited by an 82 year old man (born 1931) which has been living alone since his wife passed away two years ago. At around the same time the man had a stroke and spends most of his time inside, the exceptions are when he goes outside to do shopping or to visit any of his three sons with his mobility scooter. The man receives help from home care four times a day that ensures that he is feeling well and that he takes his medication. The man's sons live nearby and visits him often, in addition, his grandchildren uses the Giraff telepresence robot to visit him remotely.

An initial working version of all software components of the system is available in the apartment. The apartment is depicted in Fig. 3. Before deploying the system in the home the inhabitant was interviewed. The answers given during the interview was used to determine a good sensor placement that would allow the system to capture as meaningful traces of his daily activities. This resulted in the fact that the laundry room and the study were not instrumented at all

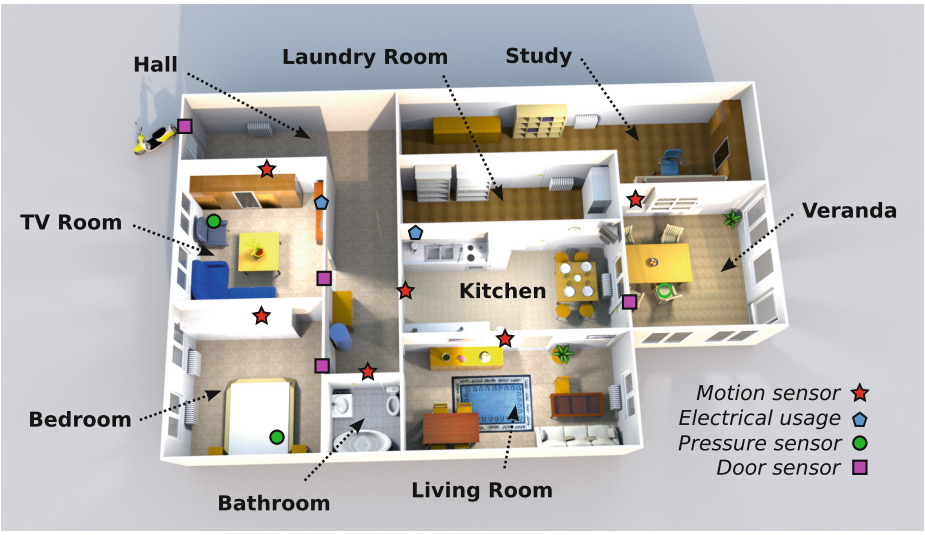


Fig. 3. The layout of the second test site in Sweden. This is not an exact depiction but captures the general layout of the large home.

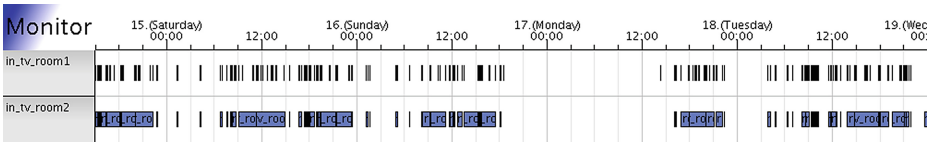


Fig. 4. A pair of timelines showing when the elderly man visits the TV room, constructed using different methods of preprocessing the sensor data.

since the inhabitant almost never used these, and the living room was sparsely instrumented since it was only used when the man had visits. Conversely, the TV-room, the kitchen, the bathroom and the bedroom were considered important and therefore equipped with more sensors.

The session with the caregiver resulted in several queries to the context recognition system using a horizon of two weeks[2]. In the beginning of the session the caregiver claimed that the man had stated that he spends much of his time in front of the TV. The caregiver wanted to know how often and when the person was watching the TV since this behavior can influence his health. Consequently, a query was made to see how much time the user spent in front of the TV using the motion sensor in the TV room[3], the output of this query is shown in Fig. 4.

The topmost timeline, `in_tv_room1`, in Fig. 4 shows the result of the first query. Given the fragmented nature of the timeline (containing many short intervals) it appeared as if the person was mostly sitting still in the TV room, or at least not moving enough to trigger the motion sensor frequently enough to generate continuous intervals on the timeline. In order to address this problem, another query was made using data from other motion sensors in the apartment as well, the output of this query is shown bottommost in Fig. 4 as `in_tv_room2`. Here, the data from the additional sensors were used as terminal conditions for ending the activity (the motion sensor placed in the hall adjacent to the TV-room was particularly important). The timeline for `in_tv_room2` is clearly more continuous than `in_tv_room1` but still contains some discontinuity. This is probably due to a bad placement of the motion sensor in the hall, allowing the user to be detected even though he is in the TV-room. At some occasions this can also be due to the fact that he had visitors, e.g. home care or relatives, as they move around the apartment they constantly end the `in_tv_room1` activity.

One responsibility of the context recognition module within GiraffPlus is to provide timelines containing performed activities to a statistics extraction module, the result of the second query forms a much better basis for assessing time spent in front of the TV during the day and can be used over longer horizons

[2] A more limited timespan was chosen for the graphics used in this paper so that details are visible.

[3] The motion sensor was used instead of the electrical usage sensor connected to the TV since the former appeared to be in an always on state. We suspect this happens because the TV consumes enough electricity in standby mode to be considered on.

to detect changes in behavior and anomalies. The rules created to detect when the person is in the TV-room is shown in Listing 1.

```
<?xml version="1.0" encoding="UTF-8" ?>
<rules home="testsite_se_2">

<preproc name="TunstallPIRSimple" in="PIR - TV Room"
    out="_in_tv_room1" args=""/>
<preproc name="TunstallPIRSimple" in="PIR - TV Room, PIR - Bedroom,
    PIR - Kitchen" out="_in_tv_room2" args=""/>

<extractor name="max" in="_in_tv_room1" out="in_tv_room1" />
<extractor name="max" in="_in_tv_room2" out="in_tv_room2" />

</rules>
```

Listing 1. A rule that infers when the person has been in the TV room using two different methods.

Even though these queries did not produce optimal visual results, the caregiver had gotten a better understanding of the persons habits, and it can clearly be seen that the person spends many hours a day in front of the TV. Also, the caregiver noted that the man's TV-watching habits were not isolated to daytime. After having inspected the man's TV-watching habits, the caregiver was interested in the evening and night time activities of the man since he could be seen to watch TV late at night at some occasions e.g. on Sunday the 16th. In addition, discussions with the person had revealed that he sometimes went up during the night to read the newspaper in the kitchen.

As the evaluation session continued the caregiver wanted to see when the person went up at night to look at the TV or to read the newspaper so rules were constructed to filter out these events. In addition to processing the sensory data, a rule that filters out events where the person had left the bed and went to either of these locations were constructed using the language of Allen's Interval Algebra. Activity intervals `awake_in_kitchen` and `awake_in_tv_room` were inferred on a timeline so that each filtered interval occurred `AFTER in_bed` and `DURING` presence at the respective locations; `in_kitchen` and `in_tv_room`. The output of this query is shown in Fig. 5.

It can be seen that the user typically visits both the TV room and the kitchen when he leaves his bed. Also, this behavior seems to be a part of a habit since it occurs so often. A fraction of the rule document created to detect when the person leaves his bed to visit the TV-room and the kitchen is shown in Listing 2.

To obtain a verification of the inferences produced by the system, the results were discussed by the elderly man. He confirmed the inferences with his own recollection of his activities. During this discussion the man expressed discomfort about the system knowing how often he had been awake during the night. Despite being well informed of the system's capabilities, he expressed that he was less comfortable with an aggregation of long term data about his habits than with alternative technologies such as observing him visually from time to time through a video camera.

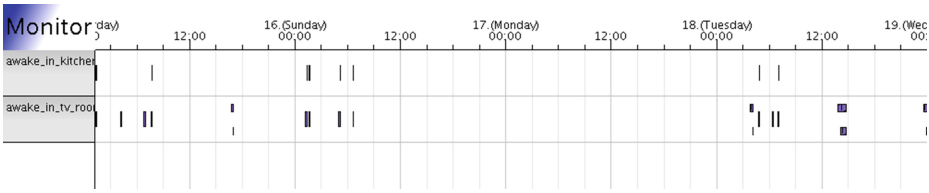


Fig. 5. A pair of timelines showing when the elderly visits the kitchen and the TV-room after having left his bed.

```
1  <?xml version="1.0" encoding="UTF-8" ?>
2  <rules home="testsite_se_2">
3
4  <preproc name="TunstallTrueFalse" in="Bed - Bedroom" out="_in_bed"/>
5  <preproc name="TunstallPIRSimple" in="PIR - Kitchen"
       out="_in_kitchen"/>
6
7  <rule out="_awake_in_kitchen">
8   <constraint from="_awake_in_kitchen" type="during"
        to="_in_kitchen"/>
9   <constraint from="_awake_in_kitchen" type="after" args="[0,1000]"
        to="_in_bed"/>
10 </rule>
11
12 <extractor name="max" in="_awake_in_kitchen"
       out="awake_in_kitchen"/>
13 ...
```

Listing 2. A fraction of a rule that is used to determine which rooms the elderly visits when he leaves his bed.

6 Conclusion

This paper has presented a fully working context recognition system that has been developed for the GiraffPlus project. The system focuses on addressing real world issues such as scarcity of sensors and the need to be able to dynamically adapt the underlying inference model to the needs of the end user and caregivers. Once fully deployed this system will analyze data coming from fifteen test sites in three different countries. In two examples we have shown how queries about the inhabitants' behavior can be customized in cooperation with a medical professional. Future research will focus on expanding the capabilities of the constraint language and evaluating its usability in the different test sites. Furthermore, we will investigate the possibility of making the collected sensor data available to the research community.

References

1. Allen, J.: Towards a general theory of action and time. Artif. Intell. **23**(2), 123–154 (1984)
2. Augusto, J., Nugent, C.: The use of temporal reasoning and management of complex events in smart homes. In: Proceedings of the 16th Eureopean Conference on Artificial Intelligence (ECAI) (2004)

3. Cesta, A., Cortellessa, G., Rasconi, R., Pecora, F., Scopelliti, M., Tiberio, L.: Monitoring elderly people with the robocare domestic environment: Interaction synthesis and user evaluation. Comput. Intell. **27**(1), 60–82 (2011). Special Issue on Scheduling and Planning Applications
4. Coradeschi, S., Cesta, A., Cortellessa, G., Coraci, L., Gonzalez, J., Karlsson, L., Furfari, F., Loutfi, A., Orlandini, A., Palumbo, F., Pecora, F., von Rump, S., Stimec, A., Ullberg, J., Östlund, B.: Giraffplus: Combining social interaction and long term monitoring for promoting independent living. In: 6th International Conference on Human System Interactions (HSI), pp. 578–585 (2013)
5. Dousson, C., Maigat, P.L.: Chronicle recognition improvement using temporal focusing and hierarchization. In: Proceedings of the 20th International Joint Conference on Artifical Intelligence, IJCAI'07, pp. 324–329. Morgan Kaufmann Publishers Inc., San Francisco (2007)
6. Duong, T., Bui, H., Phung, D., Venkatesh, S.: Activity recognition and abnormality detection with the switching hidden semi-markov model. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) (2005)
7. Goultiaeva, A., Lespérance, Y.: Incremental plan recognition in an agent programming framework. In: Working Notes of the AAAI Workshop on Plan, Activity, and Intention Recognition (PAIR) (2007)
8. Helaoui, R., Niepert, M., Stuckenschmidt, H.: Recognizing interleaved and concurrent activities: a statistical-relational approach. In: Proccedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom) (2011)
9. Jakkula, V., Cook, D., Crandall, A.: Temporal pattern discovery for anomaly detection in a smart home. In: Proceedings of the 3rd IET Conference on Intelligent Environments (IE) (2007)
10. Liao, L., Fox, D., Kautz, H.: Extracting places and activities from gps traces using hierarchical conditional random fields. Robot. Res. **26**(1), 119–134 (2007)
11. Mckeever, S., Ye, J., Coyle, L., Bleakley, C., Dobson, S.: Activity recognition using temporal evidence theory. Ambient Intell. Smart Environ. **2**(3), 253–269 (2010)
12. Modayil, J., Bai, T., Kautz, H.: Improving the recognition of interleaved activities. In: Proceedings of the 10th International Conference on Ubiquitous Computing (UbiComp) (2008)
13. Palumbo, F., Ullberg, J., Štimec, A., Furfari, F., Karlsson, L., Coradeschi, S.: Sensor network infrastructure for a home care monitoring system. Sensors 14(3), 3833–3860 (2014). http://www.mdpi.com/1424-8220/14/3/3833
14. Patterson, D., Fox, D., Kautz, H., Philipose, M.: Fine-grained activity recognition by aggregating abstract object usage. In: Proceedings of the 9th IEEE International Symposium on Wearable Computers (2005)
15. Pecora, F., Cirillo, M., Dell'Osa, F., Ullberg, J., Saffiotti, A.: A constraint-based approach for proactive, context-aware human support. J. Ambient Intell. Smart Environ. **4**(4), 347–367 (2012)
16. Pinhanez, C., Bobick, A.: Fast constraint propagation on specialized allen networks and its application to action recognition and control. Technical report 456, M.I.T. Media Lab, Perceptual Computing Section (1998)
17. Pollack, M., Brown, L., Colbry, D., McCarthy, C., Orosz, C., Peintner, B., Ramakrishnan, S., Tsamardinos, I.: Autominder: an intelligent cognitive orthotic system for people with memory impairment. Robot. Auton. Syst. **44**(3–4), 273–282 (2003)

18. Pujari, A.K., Kumari, G.V., Sattar, A.: Indu: an interval duration network. In: Foo, Norman Y. (ed.) AI 1999. LNCS, vol. 1747, pp. 291–303. Springer, Heidelberg (1999)
19. Riboni, D., Bettini, C.: Context-aware activity recognition through a combination of ontological and statistical reasoning. In: Zhang, D., Portmann, M., Tan, A.-H., Indulska, J. (eds.) UIC 2009. LNCS, vol. 5585, pp. 39–53. Springer, Heidelberg (2009)
20. Shanahan, M.: Robotics and the common sense informatic situation. In: Proceedings of the 12th European Conference on Artificial Intelligence (ECAI) (1996)
21. Singla, G., Cook, D.J., Schmitter-Edgecombe, M.: Recognizing independent and joint activities among multiple residents in smart environments. Ambient Intell. Humanized Comput. **1**(1), 57–63 (2010)
22. Springer, T., Turhan, A.Y.: Employing description logics in ambient intelligence for modeling and reasoning about complex situations. Ambient Intell. Smart Environ. **1**(3), 235–259 (2009)
23. Tazari, M.R., Furfari, F., Lázaro Ramos, J.P., Ferro, E.: The PERSONA service platform for AAL spaces. In: Nakashima, H., Aghajan, H., Augusto, J.C. (eds.) Handbook of Ambient Intelligence and Smart Environments, pp. 1171–1199. Springer, New York (2010)
24. Ullberg, J., Pecora, F.: Propagating constraints on sets of intervals. In: ICAPS Workshop on Planning and Scheduling with Timelines (PSTL) (2012)
25. Wu, J., Osuntogun, A., Choudhury, T., Philipose, M., Rehg, J.: A scalable approach to activity recognition based on object use. In: Proceedings of ICCV 2007 (2007)

The AVA Multi-View Dataset for Gait Recognition

David López-Fernández(✉), Francisco José Madrid-Cuevas, Ángel Carmona-Poyato, Manuel Jesús Marín-Jiménez, and Rafael Muñoz-Salinas

Computing and Numerical Analysis Department, Campus de Rabanales, Maimónides Institute for Biomedical Research (IMIBIC), University of Córdoba, 14071 Córdoba, Spain
{i52lofed,fjmadrid,ma1capoa,mjmarin,rmsalinas}@uco.es

Abstract. In this paper, we introduce a new multi-view dataset for gait recognition. The dataset was recorded in an indoor scenario, using six convergent cameras setup to produce multi-view videos, where each video depicts a walking human. Each sequence contains at least 3 complete gait cycles. The dataset contains videos of 20 walking persons with a large variety of body size, who walk along straight and curved paths. The multi-view videos have been processed to produce foreground silhouettes. To validate our dataset, we have extended some appearance-based 2D gait recognition methods to work with 3D data, obtaining very encouraging results. The dataset, as well as camera calibration information, is freely available for research purposes.

1 Introduction

Research on human gait as a biometric for identification has received a lot of attention due to the apparent advantage that it can be applied discreetly on the observed individuals without needing their cooperation. Because of this, the automation of video surveillance is one of the most active topics in Computer Vision. Some of the interesting applications are, among others, access control, human-machine interface, crowd flux statistics, or detection of anomalous behaviours [1].

Most current gait recognition methods require gait sequences captured from a single view, namely, from the side view or from the front view of a walking person [2–9]. Hence, there are many existing databases which capture the gait sequences from a single view. However, new challenges in the topic of gait recognition, such as achieving the independence from the camera point of view, usually require multi-view datasets. In fact, articles related to multi-view and cross-view gait recognition have been increasingly published [10–16].

First, some of the existing multi-view current datasets were recorded in controlled conditions, and in some cases, they made use of a treadmill [17–19]. An inherent problem associated with walking on a treadmill is that the human gait

P.L. Mazzeo et al. (Eds.): AMMDS 2014, pp. 26–39, 2014.
DOI: 10.1007/978-3-319-13323-2_3

is not as natural as it should be, the gait speed is usually constant, and the subjects cannot turn right or left. They are not representative of human gait in a real world. Secondly, for other multi-view datasets, calibration information is not provided, e.g. [20]. Some of the gait recognition methodologies require camera calibration to deal with 3D information.

In addition, there are not many multi-view datasets specifically designed for gait. Some of them are designed for action recognition, and therefore they do not contain gait sequences of enough length as to contain several gait cycles, because gait is a subset of them.

For this reason, we have created a new indoor dataset to test gait recognition algorithms. This dataset can be applied in workspaces where subjects cannot show the face or use the fingerprint, and even they have to wear special clothing, e.g. a laboratory. Furthermore, in this dataset people appear walking along both straight and curved paths, which makes this dataset suitable to test methods like [10]. The cameras have been calibrated and the methods based on 3D information can use this dataset to test. The dataset is free only for research purposes.

This paper is organized as follows. Section 2 describes current datasets for gait recognition. Section 3 describes the AVA Multi-View Dataset for Gait Recognition (AVAMVG). Section 4 shows several application examples carried out to validate our database. Finally, we conclude this paper in Sect. 5.

2 Current Datasets for Gait Recognition

The chronologic order of appearance of the different human action video datasets runs parallel to the challenges that the scientific community has been considering to face the problem of automatic and visual gait recognition. From this point of view, datasets can be divided into two groups: datasets for single-view gait recognition and datasets for multi-view gait recognition. Besides, the datasets can be divided in two subcategories: indoor and outdoor datasets.

Regarding single view datasets, indoor gait sequences are provided by the OU-ISIR Biometric Database [17]. OU-ISIR database is composed by four treadmill datasets, called A, B, C and D. Dataset A is composed of gait sequences of 34 subjects from side view with speed variation. The dataset B is composed of gait sequences of 68 subjects from side view with clothes variation up to 32 combinations. OU-ISIR gait dataset D contains 370 gait sequences of 185 subjects observed from the lateral view. The dataset D focuses on the gait fluctuations over a number of periods. The OU-ISIR gait dataset C is currently under preparation, and as far as we know, information about this has not been released yet.

In contrast with OU-ISIR, the first available outdoor single-view database was from the Visual Computing Group of the UCSD (University of California, San Diego) [21]. The UCSD gait database includes six subjects with seven image sequences of each, from the side view. In addition to UCSD gait database, one of the most used outdoor datasets for single-view gait recognition is the USF HumanID database [22]. This database consists of 122 persons walking in elliptical paths in front of the camera.

Other outdoor walking sequences are provided in CASIA database, from the Center for Biometrics and Security Research of the Institute of Automation of the Chinese Academy of Sciences. CASIA Gait Database is composed by three datasets, one indoor and the other two outdoor. The indoor dataset can be also considered as a multi-view dataset, and therefore it will be discussed later. The outdoors datasets are named as Dataset A and Dataset C (infrared dataset), and they are described below.

Dataset A [23] includes 20 persons. Each person has 12 image sequences, 4 sequences for each of the three directions, i.e. parallel, 45 degrees and 90 degrees with respect to the image plane. The length of each sequence is not identical due to the variation of the walker's speed. Dataset C [24] was collected by an infrared (thermal) camera. It contains 153 subjects and takes into account four walking conditions: normal walking, slow walking, fast walking, and normal walking with a bag. The videos were all captured at night.

More outdoor gait sequences are also found in the HID-UMD database [25], from University of Maryland. This database contains walking sequences of 25 people in 4 different poses (frontal view/walking-toward, frontal view/walking-away, frontal-parallel view/toward left, frontal-parallel view/toward right).

A database containing both indoor and outdoor sequences is the Southampton Human ID gait database (SOTON Database) [18]. This database consists of a large population, which is intended to address whether gait is individual across a significant number of people in normal conditions, and a small population database, which is intended to investigate the robustness of biometric techniques to imagery of the same subject in various common conditions.

Currently, in real problems, more complex situations are managed. Thus, for example, outdoor scenarios may be appropriate to deal with real surveillance situations, where occlusions occur frequently. To address the challenge of occlusions, the TUM-IITKGP Gait Database is presented in [26].

Other indoor and outdoor datasets have been specifically designed for action recognition. However, it is possible to extract a subset of gait sequences from them. Examples of this are Weizmann [27], KTH [28], Etiseo, Visor [29] and UIUC [30]. The Weizmann database contains the walking action among other 10 human actions, each action performed by nine people. KTH dataset contains six types of human actions performed several times by 25 people in four different scenarios. ETISEO and Visor were created to be applied in video surveillance algorithms. UIUC (from University of Illinois) consists of 532 high resolution sequences of 14 activities (including walking) performed by eight actors.

With the new gait recognition approaches that deal with 3D information, new gait datasets for multi-view recognition have emerged. One of the first multi-view published dataset was CASIA Dataset B [20]. Dataset B is a large multi-view gait database. There are 124 subjects, and the gait data was captured from 11 viewpoints. Neither the camera position nor the camera orientation are provided.

As can be seen in OU-ISIR and SOTON databases among others, treadmills are widely used, nonetheless, an inherent problem associated with walking on a treadmill is that the human gait is not as natural as it should be. An example

of using of the treadmill in a multiview dataset is presented in the CMU Motion of Body (MoBo) Database [19], which contains videos of 25 subjects walking on a treadmill from multiple views. A summary of the current freely available gait datasets is shown in Table 2.

Table 1. Multi-view action datasets that include walking as an action. This table shows some of the most popular multiview action datasets, which contain walking sequences among other activities.

Database	Actions	Subjects	Source	Views	Path	Year
i3DPost [31]	13	8	Indoor	Eight views	Straight	2009
MuHAVi [32]	17	14	Indoor	Eight views	Straight	2010
IXMAS [33]	11	10	Indoor	Five views	Closed curve	2006

Other multiview datasets, specifically designed for action recognition rather than gait recognition, are described then. A summary of them can be seen in Table 1. The i3DPost Multi-View Dataset [31] was recorded using a convergent eight camera setup to produce high definition multi-view videos, where each video depicts one of eight people performing one of twelve different human actions. A subset for gait recognition can be obtained from this dataset. The actors enter the scene from different entry points, which seems to be suitable to test invariant view gait recognition algorithms. However, the main drawback of this subset is the short length of the gait sequences, extracted from a bigger collection of actions.

The Faculty of Science, Engineering and Computing of Kingston University collected in 2010 a large body of human action video data named MuHAVi (Multicamera Human Action Video dataset) [32]. It provides a realistic challenge to objectively compare action recognition algorithms. There are 17 action classes (including walk and turn back) performed by 14 actors. A total of eight non-synchronized cameras are used. The main weakness of MuHAVi is that the walking activity is carried out in an unique predefined trajectory. Due to this, this dataset is not very suitable to compare invariant-view gait recognition algorithms. This dataset was specifically designed to test action recognition algorithms, and it does not contain gait sequences of enough length.

The INRIA Xmas Motion Acquisition Sequences (IXMAS) database, reported in [33], contains five-view video and 3D body model sequences for eleven actions and ten persons. A subset for gait recognition challenges can be obtained from the INRIA IXMAS database. However, humans appear walking in very closed circle paths. Consequently, the dataset does not provide very realistic gait sequences.

3 AVA Multi-view Dataset for Gait Recognition

In this section we briefly describe the camera setup, the database content, and the preprocessing steps carried out in order to further increase the applicability of the database.

Table 2. Summary of existing gait datasets. The table below shows some features of the existing gait datasets. Some of the current databases are divided into other subsets, to deal with specific challenges, as clothes variation, carrying conditions, or multiple view recognition. The column sequences shows the number of available sequences per subject.

Database	Subset	Type of problem	Subjects	Sequences	Source	Treadmill	Views	Path	Year
UCSD [21]	N.A	Shaded scenes	6	7	Outdoor	No	Side	Circular	1998
HID-UMD [25]	N.A	Undetermined	25	1	Outdoor	No	Front, side	Straight	2001
MoBo [19]	N.A	Multi-view recognition	25	4	Indoor	Yes	Six views	Straight	2001
SOTON [18]	Large	Multiple purposes	100	6	In-outdoor	Some seq.	0, 45, 90	Straight	2002
	Small	Diff. walk. cond.	12	15	Indoor	No	0, 45, 90	Straight	
CASIA	A [23]	Undetermined	20	12	Outdoor	No	0, 45, 90	Straight	2001
	B [20]	Multi-view recognition and diff. carrying cond.	124	10	Indoor	No	11 views	Straight	2005
	C [24]	Diff. walk. cond.	153	10	Outdoor	No	Side	Straight	2005
USF Human ID [22]	N.A	Covariate conditions	122	Up to 5	Outdoor	No	Side	Elliptical	2005
TUM-IITKGP [26]	N.A	Occlusions	35	1	Indoor	No	Side	Straight	2011
OU-ISIR [17]	A	Speed variation	34	68	Indoor	Yes	Side	Straight	2012
	B	Clothes variation	68	Up to 32	Indoor	Yes	Side	Straight	
	D	Gait fluctuation	370	185	Indoor	Yes	Side	Straight	
AVA	N.A	Multi-view recognition	20	10	Indoor	No	Six views	Curved and straight	2013

3.1 Studio Environment and Camera Setup

Six convergent IEEE-1394 FireFly MV FFMV-03M2C cameras are equipped in the studio where the dataset was recorded, spaced in a square of 5.8 m of side at a height of 2.3 m above the studio floor. The cameras provide 360° coverage of a capture volume of 5 m × 5 m × 2.2 m.

A natural ambient illumination is provided by four windows through which natural light enters into the scene. Video gait sequences were recorded at different times of day and the cameras were positioned above the capture volume and were directed downward.

Instead of using a screen backdrop of a specific color, as in [31], the background of the scene is the white wall of the studio. However, to facilitate foreground segmentation, the actors wear clothes of different color than the background scene.

Human gait is captured in 4:3 format with 640 × 480 pixels at 25 Hz. Synchronized videos from all six cameras were recorded uncompressed directly to disk with a dedicated PC capture box. All cameras were calibrated to extract their intrinsic (focal length, centre of projection, and distortion coefficients) and extrinsic (pose, orientation) parameters.

To get the intrinsics of each camera, we used a classical black-white chessboard based technique [34] (OpenCV), while for the extrinsics we used the Aruco library [35] whose detection of boards (several markers arranged in a grid) have two main advantages. First, since there is more than one marker, it is less likely to lose them all at the same time. Second, the more markers detected, the more points available for computing the camera extrinsics. An example of extrinsics calibration based on Aruco library is shown in Fig. 1. Calibration of the studio multi-camera system can be done in less than 10 min using the above referenced techniques.

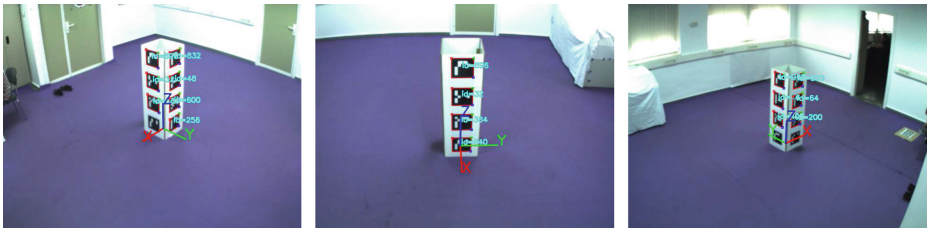


Fig. 1. 3D artifact with Aruco [35] **board of markers**, used for getting the pose and orientation of each camera.

3.2 Database Description

Using the camera setup described above, twenty humans (4 females and 16 males), participated in ten recording sessions each. Consequently, the database contains 200 multi-view videos or 1200 (6 × 200) single view videos. In the following paragraphs we describe the walking activity carried out by each actor of the database.

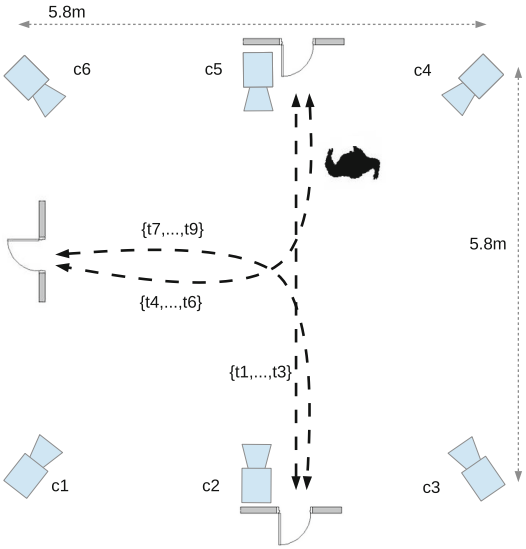


Fig. 2. Workspace setup for dataset recording, where $\{c1, ..., c6\}$ represent the set of cameras of the multiview dataset and $\{t1, ..., t9\}$ represent the different trajectories followed by each actor of the dataset.

Ten gait sequences were designed before the recording sessions. All actor depict three straight walking sequences ($\{t1, ..., t3\}$), and six curved gait sequences ($\{t4, ..., t9\}$), as if they had to turn a corner. The curved paths are composed by a first section in straight line, then a slight turn, and finally a final straight segment. These paths are graphically described in Fig. 2. In the last sequence actors describe a figure-eight path ($t10$).

3.3 Multi-view Video Preprocessing

The raw video sequences were preprocessed to further increase the applicability of the database. To obtain the silhouettes of actors, we have used the Horprasert's algorithm [36]. This algorithm is able to detect moving objects in a static background scene that contains shadows on color images, and it is also able to deal with local and global perturbations such as illumination changes, casted shadows and lightening.

In Fig. 3, several walking subjects of the AVA Multi-View Dataset for Gait Recognition are shown.

4 Database Application Examples

In this section, we carry out several experiments to validate our database. First, we use a Shape from Silhouette algorithm [37] to get 3D reconstructed human

Fig. 3. Example of our multiview dataset. People walking in different directions, from multiple points of view.

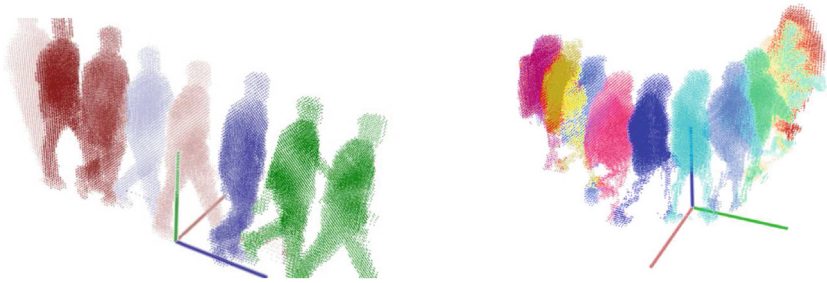


Fig. 4. 3D reconstructed gait sequences. Example of reconstructed gait sequences, sampled at 2 Hz, where each point represents the center of a squared voxel.

volumes along the gait sequence. The whole gait sequences can be reconstructed, as shown in Fig. 4. Then, these gait volumes are aligned and centred respect to a global reference system. After this, we can get rendered projections of these volumes to test 2D-based gait recognition algorithms. By this way, we can test view-dependent gait recognition algorithms on any kind of path, either curved or straight.

4.1 Gait Recognition Based on Rendered Gait Entropy Images

One of the most cited silhouette-based representations is the Gait Energy Image (GEI) [9], which represents gait using a single grey scale image obtained by averaging the silhouettes extracted over a complete gait cycle. In addition to GEI, a gait representation called Gait Entropy Image (GEnI) is proposed in [7]. GEnI encodes in a single image the randomness of pixel values in the silhouette images over a complete gait cycle. Thus it is a compact representation which is an ideal starting point for feature selection. In fact, GEnI was proposed to measure the relevance of gait features extracted from the GEI.

As we have aligned gait volumes, we can use rendered side projections of the aligned volumes to compute the GEI. In addition, the GEnI can be computed by calculating Shannon entropy for each pixel of the silhouette images, rendered from side projections of the aligned volumes. In this way, the methods proposed in [7,9] can be tested in a view invariant way. Figure 5 shows the GEI and GEnI descriptors computed over rendered images of the aligned sequence.

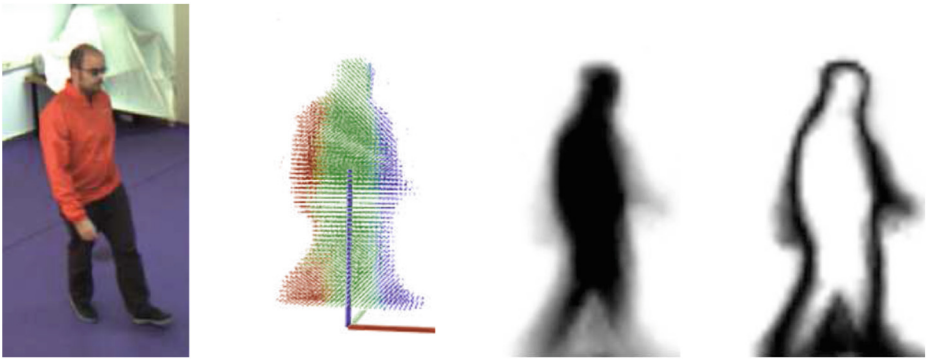


Fig. 5. GEI and GEnI. The leftmost image shows a walking subject. The reconstructed volumes are aligned along the gait sequence, as can be seen in the second image. The two last images show the GEI and GEnI computed over rendered images of the aligned sequence, respectively.

We designed a hold-out experiment where the gallery set is composed by the 1st, 2nd, 4th, 5th, 7th and 8th sequences and probe set is formed by 3rd, 6th, and 9th sequences of the AVA Multi-View dataset. The recognition rate

obtained with the application of the gait descriptors proposed in [7,9] are shown in Table 3.

Table 3. Results of the algorithm proposed in [7] on the AVA Multi-View dataset, based on silhouettes. We report the recognition rate in %, comparing GEI with the GEnI by direct template matching, using the AVA Multiview Dataset.

Database (probe)	GEI	GEnI
AVA Multiview Dataset (AVA B)	94.6	98.1

4.2 Front View Gait Recognition by Intra- and Inter-frame Rectangle Size Distribution

In [8], video cameras are placed in hallways to capture longer sequences from the front view of walkers rather than the side view, which results in more gait cycles per gait sequence. To obtain a gait representation, a morphological descriptor, called Cover by Rectangles (CR), is defined as the union of all the largest rectangles that can fit inside a silhouette. Despite of the high recognition rate, a drawback of this approach is the dependence with respect to the angle of the camera.

According to the authors, Cover by Rectangles has the following useful properties: (1) the elements of the set overlap each other, introducing redundancy (i.e. robustness), (2) each rectangle covers at least one pixel that belongs to no other rectangle, and (3) the union of all rectangles reconstructs the silhouette so

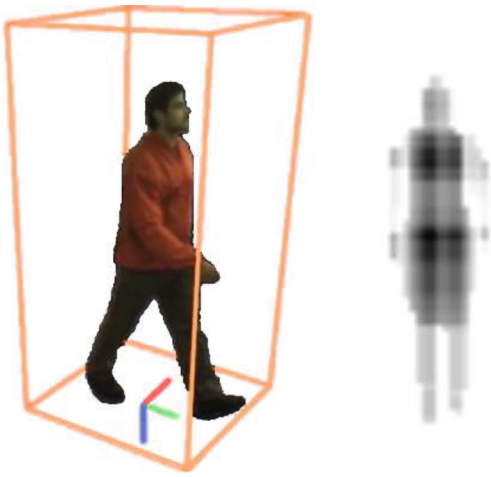


Fig. 6. Cover by Rectangles descriptor. Bounding box of a walking human (left), Cover by Rectangles descriptor (right). A gray level on pixel displays the density of rectangles that contains that pixel.

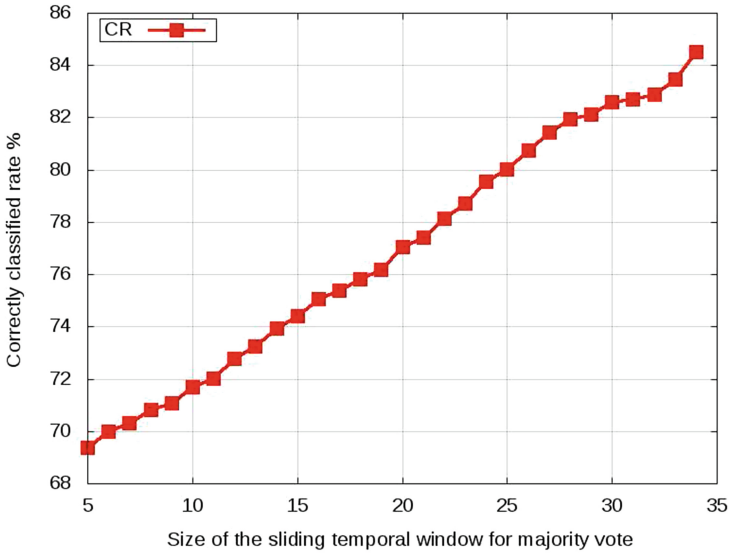


Fig. 7. Recognition rate obtained with the application of the appearance based algorithm proposed in [8]. Since we have aligned the reconstructed volumes along the gait sequence, we can use a frontal projection of them. We show the effect on the classification rate of using a sliding temporal window for voting.

that no information is ever lost. A representation of this descriptor can be seen in Fig. 6.

As we have aligned gait volumes, we can use front-rendered projections of the aligned volumes to compute the CR, and therefore the method proposed in [8] can be tested in a view invariant way.

To test the algorithm with the AVA Multi-view Dataset, we use a leave-one-out cross-validation. Each fold is composed by a tuple formed by a set of 20 sequences (one sequence per actor) for testing, and by the remaining eight sequences of each actor for training, i.e. 160 sequences for training and 20 sequences for test. We use SVM with Radial Basis Functions, since we obtained better results than with others classifiers. To make the choice of SVM parameters independent of the sequence test data, we cross-validate the SVM parameters on the training set. For this experiment, the features vector size was set to $L = 20$, and the histogram size with which the highest classification rate is achieved is $M = N = 25$ (see [8]). With the CR descriptor applied on the frontal volume projection, we obtain a maximum accuracy of 84.52 %, as can be seen in Fig. 7.

5 Conclusions

In this paper, we present a new multi-view database containing gait sequences of 20 actors that depict ten different trajectories each. The database has been

specifically designed to test multi-view and 3D based gait recognition algorithms. The dataset contains videos of 20 walking persons (men and women) with a large variety of body size, who walk along straight and curved paths. The cameras have been calibrated and both calibration information and binary silhouettes are also provided.

To validate our database, we have carried out some experiments. We began with the 3D reconstruction of volumes of walking people. Then, we aligned and centred them respect to a global reference system. After this, since we have reconstructed and aligned gait sequences, we used rendered projections of these volumes to test some appearance-based algorithms that work with silhouettes to identify an individual by his manner of walking.

This dataset can be applied in workspaces where subjects cannot show the face or use the fingerprint, and even they have to wear special clothing, e.g. a laboratory. The dataset is free only for research purposes[1].

Acknowledgements. This work has been developed with the support of the Research Projects called TIN2012-32952 and BROCA both financed by Science and Technology Ministry of Spain and FEDER.

References

1. Hu, W., Tan, T., Wang, L., Maybank, S.: A survey on visual surveillance of object motion and behaviors. IEEE Trans. Syst. Man Cybern. **34**, 334–352 (2004)
2. Lee, C.P., Tan, A.W.C., Tan, S.C.: Gait recognition via optimally interpolated deformable contours. Pattern Recogn. Lett. **34**, 663–669 (2013)
3. Das Choudhury, S., Tjahjadi, T.: Gait recognition based on shape and motion analysis of silhouette contours. Comput. Vis. Image Underst. **117**, 1770–1785 (2013)
4. Zeng, W., Wang, C.: Human gait recognition via deterministic learning. Neural Netw. **35**, 92–102 (2012)
5. Roy, A., Sural, S., Mukherjee, J.: Gait recognition using pose kinematics and pose energy image. Sig. Process. **92**, 780–792 (2012)
6. Huang, X., Boulgouris, N.: Gait recognition with shifted energy image and structural feature extraction. IEEE Trans. Image Process. **21**, 2256–2268 (2012)
7. Bashir, K., Xiang, T., Gong, S.: Gait recognition without subject cooperation. Pattern Recogn. Lett. **31**, 2052–2060 (2010)
8. Barnich, O., Van Droogenbroeck, M.: Frontal-view gait recognition by intra- and inter-frame rectangle size distribution. Pattern Recogn. Lett. **30**, 893–901 (2009)
9. Han, J., Bhanu, B.: Individual recognition using gait energy image. IEEE Trans. Pattern Anal. Mach. Intell. **28**, 316–322 (2006)
10. Iwashita, Y., Ogawara, K., Kurazume, R.: Identification of people walking along curved trajectories. Pattern Recogn. Lett. **48**(0), 60–69 (2014)
11. Kusakunniran, W., Wu, Q., Zhang, J., Li, H.: Gait recognition under various viewing angles based on correlated motion regression. IEEE Trans. Circuits Syst. Video Technol. **22**, 966–980 (2012)

[1] Full database access information: http://www.uco.es/grupos/ava/node/41.

12. Krzeszowski, T., Kwolek, B., Michalczuk, A., Świtoński, A., Josiński, H.: View independent human gait recognition using markerless 3D human motion capture. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) ICCVG 2012. LNCS, vol. 7594, pp. 491–500. Springer, Heidelberg (2012)
13. Lu, J., Tan, Y.P.: Uncorrelated discriminant simplex analysis for view-invariant gait signal computing. Pattern Recogn. Lett. **31**, 382–393 (2010)
14. Goffredo, M., Bouchrika, I., Carter, J., Nixon, M.: Self-calibrating view-invariant gait biometrics. IEEE Trans. Syst. Man Cybern. Part B: Cybern. **40**, 997–1008 (2010)
15. Kusakunniran, W., Wu, Q., Li, H., Zhang, J.: Multiple views gait recognition using view transformation model based on optimized gait energy image. In: 2009 IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops), pp. 1058–1064 (2009)
16. Bodor, R., Drenner, A., Fehr, D., Masoud, O., Papanikolopoulos, N.: View-independent human motion classification using image-based reconstruction. Image Vis. Comput. **27**, 1194–1206 (2009)
17. Makihara, Y., Mannami, H., Tsuji, A., Hossain, M., Sugiura, K., Mori, A., Yagi, Y.: The ou-isir gait database comprising the treadmill dataset. IPSJ Trans. Comput. Vis. Appl. **4**, 53–62 (2012)
18. Shutler, J., Grant, M., Nixon, M.S., Carter, J.N.: On a large sequence-based human gait database. In: Proceedings of RASC, pp. 66–72. Springer (2002)
19. Gross, R., Shi, J.: The cmu motion of body (mobo) database. Technical report CMU-RI-TR-01-18, Robotics Institute, Pittsburgh, PA (2001)
20. Yu, S., Tan, D., Tan, T.: A framework for evaluating the effect of view angle, clothing and carrying condition on gait recognition. In: 18th International Conference on Pattern Recognition, ICPR 2006, vol. 4, pp. 441–444 (2006)
21. Nixon, M.S., Tan, T.N., Chellappa, R.: Human Identification Based on Gait, vol. 4. Springer, New York (2006)
22. Sarkar, S., Phillips, P.J., Liu, Z., Vega, I.R., Grother, P., Bowyer, K.W.: The humanid gait challenge problem: data sets, performance, and analysis. IEEE Trans. Pattern Anal. Mach. Intell. **27**, 162–177 (2005)
23. Wang, L., Tan, T., Ning, H., Hu, W.: Silhouette analysis-based gait recognition for human identification. IEEE Trans. Pattern Anal. Mach. Intell. **25**, 1505–1518 (2003)
24. Tan, D., Huang, K., Yu, S., Tan, T.: Efficient night gait recognition based on template matching. In: 18th International Conference on Pattern Recognition, ICPR 2006, vol. 3, pp. 1000–1003 (2006)
25. Chalidabhongse, T., Kruger, V., Chellappa, R.: The umd database for human identification at a distance. Technical report, University of Maryland (2001)
26. Hofmann, M., Sural, S., Rigoll, G.: Gait recognition in the presence of occlusion: a new dataset and baseline algorithm. In: Proceedings of 19th International Conference on Computer Graphics, Visualization and Computer Vision (WSCG), Plzen, Czech Republic, 31 January 2011–03 February 2011
27. Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. In: Tenth IEEE International Conference on Computer Vision, ICCV 2005, vol. 2, pp. 1395–1402 (2005)
28. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local svm approach. In: Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004, vol. 3, pp. 32–36 (2004)
29. Vezzani, R., Cucchiara, R.: Video surveillance online repository (visor): an integrated framework. Multimedia Tools Appl. **50**, 359–380 (2010)

30. Tran, D., Sorokin, A.: Human activity recognition with metric learning. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 548–561. Springer, Heidelberg (2008)
31. Gkalelis, N., Kim, H., Hilton, A., Nikolaidis, N., Pitas, I.: The i3dpost multi-view and 3d human action/interaction database. In: Proceedings of the 2009 Conference for Visual Media Production, CVMP '09, pp. 159–168. IEEE Computer Society, Washington, DC (2009)
32. Singh, S., Velastin, S., Ragheb, H.: Muhavi: a multicamera human action video dataset for the evaluation of action recognition methods. In: 2010 Seventh IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 48–55 (2010)
33. Weinland, D., Ronfard, R., Boyer, E.: Free viewpoint action recognition using motion history volumes. Comput. Vis. Image Underst. **104**, 249–257 (2006)
34. Bradski, G., Kaehler, A.: Learning OpenCV: Computer Vision with the OpenCV Library. O'Reilly, Cambridge (2008)
35. Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F., Marín-Jiménez, M.: Automatic generation and detection of highly reliable fiducial markers under occlusion. Pattern Recogn. **47**, 2280–2292 (2014)
36. Horprasert, T., Harwood, D., Davis, L.S.: A statistical approach for real-time robust background subtraction and shadow detection. In: Proceedings of IEEE ICCV, pp. 1–19 (1999)
37. Díaz-Más, L., Muñoz-Salinas, R., Madrid-Cuevas, F., Medina-Carnicer, R.: Shape from silhouette using dempster shafer theory. Pattern Recogn. **43**, 2119–2131 (2010)

Topological Features for Monitoring Human Activities at Distance

Javier Lamar Leon[1(✉)], Raúl Alonso[1], Edel Garcia Reyes[1], and Rocio Gonzalez Diaz[2]

[1] Patterns Recognition Department, Advanced Technologies Application Center, 7th A # 21406 e/ 214 y 216, Rpto. Siboney, CP 12200 Playa, La Habana, Cuba
{jlamar,rbaryolo,egarcia}@cenatav.co.cu
[2] Applied Math Department, School of Computer Engineering, Campus Reina Mercedes, University of Seville, Seville, Spain
rogodi@us.es

Abstract. In this paper, a topological approach for monitoring human activities is presented. This approach makes possible to protect the person's privacy hiding details that are not essential for processing a security alarm. First, a stack of human silhouettes, extracted by background subtraction and thresholding, are glued through their gravity centers, forming a 3D digital binary image I. Secondly, different orders of the simplices are applied on a simplicial complex obtained from I, which capture relations among the parts of the human body when walking. Finally, a *topological signature* is extracted from the persistence diagrams according to each order. The measure cosine is used to give a similarity value between topological signatures. In this way, the powerful topological tool known as persistent homology is novelty adapted to deal with gender classification, person identification, carrying bag detection and simple action recognition. Four experiments show the strength of the topological feature used; three of they use the CASIA-B database, and the fourth use the KTH database to present the results in the case of simple actions recognition. In the first experiment the named topological signature is evaluated, obtaining 98.8 % (lateral view) of correct classification rates for gender identification. In the second one are shown results for person identification, obtaining an average of 98.5 %. In the third one the result obtained is 93.8 % for carrying bag detection. And in the last experiment the results were 97.7 % walking and 97.5 % running, which were the actions took from the KTH database.

Keywords: Gait-based recognition · Topology · Persistent homology · Gender classification · Carrying bag detection · Action recognition

1 Introduction

Objects detected by a video surveillance system are usually classified into different categories: human, vehicle, animal, etc. In the case of persons, it is useful another

P.L. Mazzeo et al. (Eds.): AMMDS 2014, pp. 40–51, 2014.
DOI: 10.1007/978-3-319-13323-2_4

level of categorization, which gives clues for the interpretation. After finding the class to which an object belongs, one may try to identify it and interpret its individual behavior in the scene, as well as its interaction with other objects. We consider that nonrigid objects and its actions should be described by the dynamic spatial relations among its different parts.

Methods based on geometric features extracted from silhouettes or its contours [1] have been widely used for gait recognition tasks. However, the stability of such features is affected by deformations in the shape of the silhouette. Even for the same individual, little changes on the walking direction, illumination variations and the way the clothes fit to the human body, may cause variability on the geometric features. We conjecture that topological descriptions based on the persistence of homology classes are more invariant to changes and noise in the silhouette shape than classical approaches. This kind of features have been previously used to match nonrigid shapes [2,3], because they are invariant under continuous deformations of the object.

Homology is a topological invariant frequently used in practice [4,5]. The ranks of the homology groups, also called Betti numbers, coincide in the first three dimensions with the number of connected components, tunnels and cavities of the object respectively, this as a consequence of the *Alexander duality* [6]. In particular, the homology could be a robust representation, because the shape of connected components and holes may change under geometric transformations, but their amount will be more stable. Given that is not enough to reach the invariance for the representation, but also needed a set of discriminating features, the approach called homological persistence, which is introduced from now on, will be used in order to elevate the discriminating power of the representation.

A *k-simplex* σ in $\mathbf{R^d}$ is the convex hull of a set S of $k+1$ affinely independent points, where $0 \leq k \leq d$. The dimension of σ is $dim(\sigma) = |S|-1 = k$. In (Fig. 1a) from left to right are shown k-simplices of dimensions 0, 1 and 2, which are the only used in this work. For every $U \subseteq S$ the simplex σ' defined by U is said to be a *face* of σ, if $U \neq S$ then σ' is a *proper face* of σ. A simplex σ is a *facet* or *coface* of a simplex σ' when σ' is a face of σ.

Let K be a collection of simplices, K is a *simplicial complex* if it satisfies two properties, namely (i) if σ' is a face of σ and $\sigma \in K$ then $\sigma' \in K$, and (ii) if

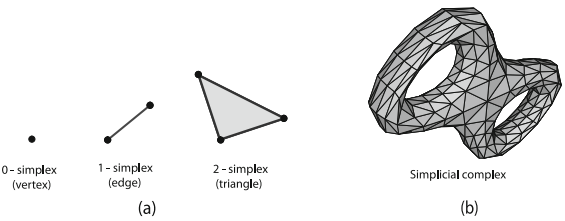


Fig. 1. (a) Simplices of dimensions 0, 1 and 2 (left to right), (b) Simplicial complex using 2-simplices (triangles) as maximal dimension simplices.

$\sigma_1, \sigma_2 \in K$ then $\sigma_1 \cap \sigma_2$ is empty or a face of both (Fig. 1b). The *dimension* of K is the largest dimension of any of its simplices. It is important to point out that a subset L of K is a *subcomplex* of K if it is a complex itself, i.e., L is a subcomplex of K iff it satisfy property (i), note that L inherit (ii) from K.

Let K be a simplicial complex with n simplices, a *filter function* $f : K \to \mathbf{R}$ is a function that assigns a real number to each simplex in K. A *filter* of K is an ordering of its simplices $[\sigma_1, \sigma_2, ..., \sigma_n]$, which satisfy that each prefix $K_i = [\sigma_1, \sigma_2, ..., \sigma_i]$ is a subcomplex of the next prefix K_{i+1}. Persistent homology concerns, given a filter, with how long persist homology classes (connected components, tunnels, cavities, etc.) after they are born; note that looking to a filter as a growing simplicial complex, we may see that homology classes are born and die. The homology classes that persist throughout all the filter i.e., that are born and never die, are the homology classes of K. An algorithm to compute persistent homology takes as input a filter, and gives out a collection of pairs representing the birth and death time of the homology classes. The difference between the birth and death time of a homology class is called its *persistence*.

The persistence diagrams and barcodes are two ways of representing the collection of pairs given up by the algorithm, they are used to study and visualize the persistent homology. In (Fig. 2 [7]) is shown an example of barcode representation, where the start of a horizontal bar represents the birth of a homology class and the end its death. An easy handling matlab implementation of an algorithm for computing persistent homology can be found[1].

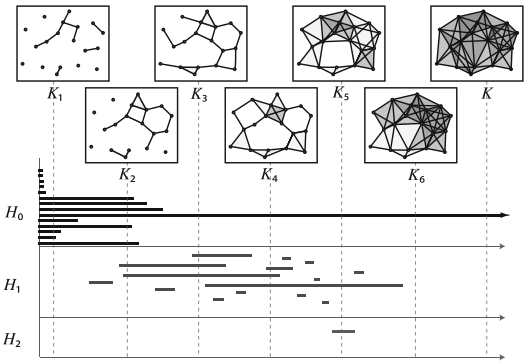


Fig. 2. Persistent barcodes. K_5 betti numbers are $H_0 = 1, H_1 = 1, H_2 = 1$ and K_2 betti numbers are $H_6 = 1, H_2 = 1, H_0 = 1$. Took from [7].

In this paper, a version of the topological features based on homological persistence given in [8,9] is presented, which is valid for gender classification, gait based person identification, carrying bag detection and simple action recognition. As aforementioned, monitoring these human activities makes possible to protect the person's privacy hiding details, such as faces, that are not essential for

[1] http://comptop.stanford.edu/programs/plex-2.0.1-windows.zip

processing a security alarm. The topological features can be used for monitoring human activities even when is possible to use images of high resolution.

The rest of the paper is organized as follows. Section 2 is devoted to describe the method for obtaining the topological signature. Experimental results are then reported in Sect. 3. We conclude this paper and discuss some future work in Sect. 4.

2 Topological Signature for Activities Monitoring

In this section, a topological signature is presented that is used for gender classification, person identification, carrying bag detection and simple action recognition at distance. A traditional approach for constructing a simplicial complex, departs from a points cloud, for which it is necessary to recover topological relations among its points. In our case there are structural relations among the points (pixels), and in fact there are temporal relation too, given that we work with videos. We take advantage of this relations to construct the simplicial complex.

2.1 The Simplicial Complex $\partial K(I)$

First, the foreground (person) is segmented from the background by applying background subtraction and thresholding. The sequence of resulting silhouettes is analyzed to extract one *subsequence of representation*, which includes at least a gait cycle [10].

The 3D digital binary image $I = (\mathbb{Z}^3, B)$ (where $B \subset \mathbb{Z}^3$ is the foreground), is built by stacking silhouettes of a subsequence of representation, aligned by their gravity centers (gc), see (Fig. 3a) and (Fig. 3b). The 3D cubical complex $Q(I)$ associated to I contains the unit cubes with vertices $V = \{(i, j, k), (i + 1, j, k), (i, j + 1, k), (i, j, k + 1), (i + 1, j + 1, k), (i + 1, j, k + 1), (i, j + 1, k + 1), (i + 1, j + 1, k + 1)\}$ and all its faces (vertices, edges and squares) iff $V \subseteq B$.

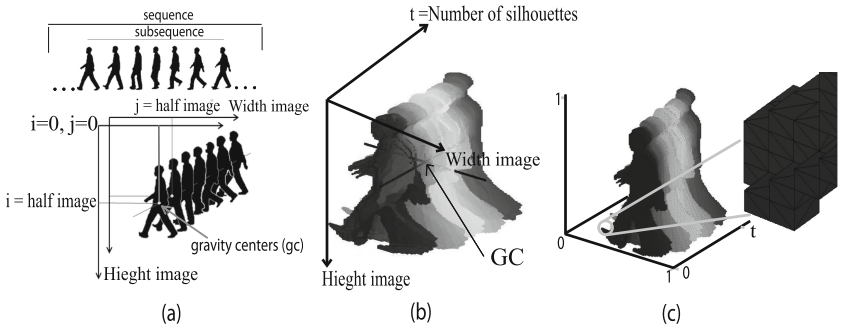


Fig. 3. (a) Silhouettes aligned by their gravity centers. (b) $I = (\mathbb{Z}^3, B)$ obtained from the silhouettes (GC is the gravity center of I). (c) The border simplicial complex $\partial K(I)$.

The squares of $Q(I)$ that are faces of exactly one cube in $Q(I)$ are subdivided in two triangles. The obtained triangles and their faces (vertices and edges) make up the *border simplicial complex* $\partial K(I)$ (see Fig. 3c). Finally, coordinates of the vertices of $\partial K(I)$ are *normalized* to coordinates (x, y, t), where $0 \leq x, y \leq 1$ and t is the number of silhouette of the subsequence of representation.

2.2 Filters for $\partial K(I)$

The topology of $\partial K(I)$ is, in general, very poor. However, in this and the next subsections we present how, using persistence diagrams, it is possible to get a topological signature from $\partial K(I)$ that captures relations among the parts of the human body when walking, and is robust against small input-data perturbations.

Up to medical researches [11,12], the natural human gait is defined as a succession of rhythmic and alternate movements of the limbs and the torso. Therefore, the difference in the gait is given by the relative position of the limbs and torso in each moment (structural features). A growing scheme of the simplicial complex $\partial K(I)$ respect to a useful selected filtration can encode the relations among the parts of the human body, which can be therefore very discriminating.

When a view direction d is chosen, two filters for $\partial K(I)$ are obtained as follows. All simplices belonging to $\partial K(I)$ are associated with two filter functions f_+ and f_-. For each vertex $v \in \partial K(I)$, $f_+(v)$ is the distance between v and the plane normal to d passing through the origin of the reference frame, while $f_-(v) = -f_+(v)$. Edges and triangles are associated to the smallest value that f_+ (resp. f_-) assumes on their vertices. Being the simplices of $\partial K(I)$ finite in number, we can determine a minimum value for f_+, say $f_{\min}$, and a maximum one, $f_{\max}$. It is now possible to induce two filters on $\partial K(I)$ by ordering its simplices according to increasing values of f_+ and f_- respectively, or according to increasing dimension of the simplices in case of tie, or arbitrarily otherwise. Denote these filters by $K_{[f_{\min}, f_{\max}]} = [\sigma_1, \ldots, \sigma_k]$ and $K_{[-f_{\max}, -f_{\min}]} = [\sigma'_1, \ldots, \sigma'_k]$.

2.3 Persistence Diagrams and Topological Signatures

Given a simplicial complex K, a filter function f, and the corresponding filter $[\sigma_1, \ldots, \sigma_k]$ for K, if σ_i completes a $p-$cycle (p is the dimension of σ_i) when σ_i is added to $K_{i-1} = [\sigma_1, \ldots, \sigma_{i-1}]$ then a p-homology class γ *is born at time* i, otherwise, a $(p-1)$-homology class *dies at time* i. The difference between the birth and death time of a homology class is known as its persistence, which quantifies the significance of a topological attribute. If γ never dies, we set its persistence to infinity. Drawing an horizontal segment $[i, j)$, in a 2D plane, for a p-homology class that is born at time i and dies at time j, we get the p-barcode diagram of the filtration. It represents a p-homology class by a segment whose length is the persistence of that class.

In this paper, barcodes are first computed for $K_{[f_{\min}, f_{\max}]}$ and $K_{[-f_{\max}, -f_{\min}]}$. Then, the barcodes are explored according to a uniform sampling. More precisely, given an integer $n > 0$, $n-1$ cuts are performed homogeneously in the complex $K_{[f_{\min}, f_{\max}]}$(resp. $K_{[-f_{\max}, -f_{\min}]}$) as follows: Let's suppose $K_{[f_{\min}, f_{\max}]} = \{\sigma_0, \ldots, \sigma_m\}$

and let's take $P_i = [\sigma_{\lfloor \frac{(i-1)m}{n} \rfloor + 1}, \ldots, \sigma_{\lfloor \frac{im}{n} \rfloor - 1}]$, $1 \leq i \leq n$, as the partitions given by the cuts. For a fixed i, we compute:

(a) Number of homology classes that were born or persist when the simplex $\sigma_{\lfloor \frac{(i-1)m}{n} \rfloor}$ is added, and, persist or die when the simplex $\sigma_{\lfloor \frac{im}{n} \rfloor}$ is added.
(b) Number of homology classes that were born in P_i.

An analogous process is done for $K_{[-f_{\max}, -f_{\min}]}$. A vector of $2n$ entries is then formed containing (a) in entry $2i$ and (b) in $2i+1$; this way we obtain for a given dimension p, a vector for each filter of K.

The *topological signature for a gait subsequence considering a fixed direction of view* consists in four $2n$-dimensional vectors: (V_1, V_2, V_3, V_4) constructed as explained above. Consider that we take into account two dimensions ($p = 0$ and $p = 1$) and two filters ($K_{[f_{\min}, f_{\max}]}$ and $K_{[-f_{\max}, -f_{\min}]}$).

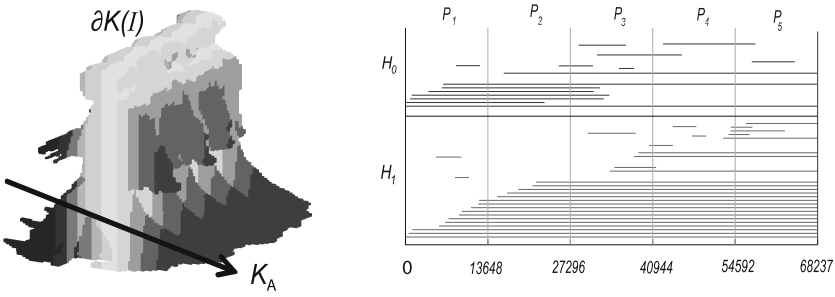


Fig. 4. An example of computation of the first element of a topological signature.

For example, consider $\partial K(I)$ given in (Fig. 4) and the direction of view K_A. We perform 4 uniform cuts on $\partial K(I)_{[f_{\min}, f_{\max}]}$, see the green lines in the persistence barcode representation. Let's fix $i = 2$ and dimension $d = 0$, according to Fig. 4, the number of homology classes that persist or were born in σ_{13648}, and, persist or die in σ_{27296} are $H_0 = 6$ in dimension 0 and the number of the homology classes that were born in P_2 are $H_0 = 2$ in dimension 0.

2.4 Comparing Topological Signatures

The topological signatures for two gait subsequences associated with a fixed view direction, say $V = \{V_1, \ldots, V_4\}$ and $W = \{W_1, \ldots, W_4\}$, can be compared according to the following procedure: for every $i = \{1, \ldots, 4\}$ compute:

$$S_i = \frac{V_i \cdot W_i}{\|V_i\| \cdot \|W_i\|}. \tag{1}$$

which is the cosine of the angle between the vectors V_i and W_i. Observe that $0 \leq S_i \leq 1$ since the entries of both vectors are always non-negative. Then, the

total similarity value for two gait subsequences, O_1 and O_2, considering a fixed view direction, is the sum of the 4 similarity measures computed before:

$$S(O_1, O_2) = S_1 + S_2 + S_3 + S_4. \quad (2)$$

3 Experimental Results

3.1 Human Gender Classification

Human gender classification can be obtained based on face [13], voice [14] or gait [15,16]. Dynamic features when the person walks give the possibility to classify gender at a distance, without any interaction with the subject [10,17,18]. This fact can improve the performance of intelligent surveillance systems and it can reduce the false positive rate during re-identification of an individual on a wide network camera. People not only observe the global motion properties while human walks, but they detect motion patterns of local body parts. For instance, women tent to swing their hips more than their shoulders. On the contrary, men tent to swing their shoulders more than their hips [19]. Moreover, men have in general wider shoulders than women [20].

Experiment 1. The performance of the proposed method is evaluated using the lateral view (90 degrees respect to the camera) in CASIA-B database. This database is composed of 124 subjects, 92 men and 32 women. For each person in the database there are 6 walking sequences, each one provided with background subtraction.

In order to avoid bias we selected 25 men and 25 women to perform the experiment. The 50 subjects were divided in 25 disjoint sets, each one containing two subjects (one man and one woman). Only one of these 25 sets was used for testing. The remaining 24 sets were used for training. The correct classification rate (CCR) is the average of the 25 combination of the cross validation.

The experimental protocol was made according to [15,16]. In this experiment, a subsequence of representation corresponds to the whole sequence, which has two gait cycles as average. We fixed $n = 24$ and used 3 view directions. The first one is vertical (i.e. parallel to axis y). The second one forms 45 degrees with axes x and y and 90 degrees with axis t. The third one is parallel to axis t, (see Fig. 5). In each experiment, the results of our method are compared with the methods presented in [15,16].

The aim of this experiment is to evaluate the topological signature for gender classification. Table 1 shows the 25-fold-cross-validation of CCR for the whole body using lateral view, as aforementioned. We can see that the topological signature provides better results.

3.2 Gait Recognition

Gait recognition is a challenging problem that gives the possibility to identify persons at a distance without any interaction with the subjects, which is very important in real surveillance scenarios [10,18].

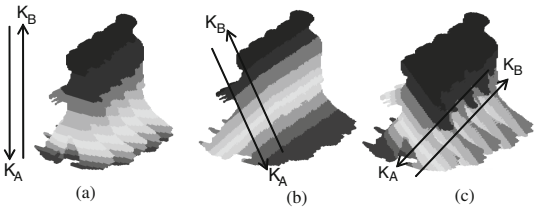


Fig. 5. View directions used in the experiments.

Table 1. Correct classification rates for gait based gender classification.

Method	Avg computer [15]	Avg human observers [15]	MCRF [16]	Our method
CCR	95.97	95.45	98.3	98.88

Experiment 2. We show the performance of the proposed method on the 11 views of the CASIA-B database, which contains 124 subjects. In this experiment the subsequence of representation consists of all the sequence, and the n parameter was set to 24.

We have used 4 view directions, they are shown in (Fig. 6) by black segments going through the simplicial complex and intercepting the GC point. The first one is parallel to the y axis and perpendicular to x and t, (Fig. 6a). The second one is parallel to the x axis and perpendicular to y and t, (Fig. 6b). The third one forms 45 degrees with the x and y axes and is perpendicular to t, (Fig. 6c). And the last is perpendicular to the previous one, perpendicular to the t axis and as aforementioned goes through the GC point. In all the cases in (Fig. 6) is shown the simplicial complex interleaved between two planes orthogonal to the view direction.

The experiment was carried out using 4 video sequences for training, and 2 for testing. The results are compared with the ones in [1]. In Table 2 is shown the cross validation average (15 combinations) of correct classification rates. It can be seen in Table 2 that the topological approach has a better performance for almost all the view angles, but the algorithm developed in [1] performs better for view angles close to 0 degrees. Therefore, these complementary behaviors conduce us to think in a combination of both approaches in future works.

3.3 Carrying Bag Detection

When the goal is to detect a simple behavior, namely a person has left a bag somewhere, it may be important to know that the person is carrying a bag. As carrying some object changes the normal body movements, topological features could be used to differentiate if the person left the bag [21–23].

Experiment 3. In this experiment we used, once again, the CAISA-B database. From the 124 persons, we selected 100 persons carrying bag and 100 walking

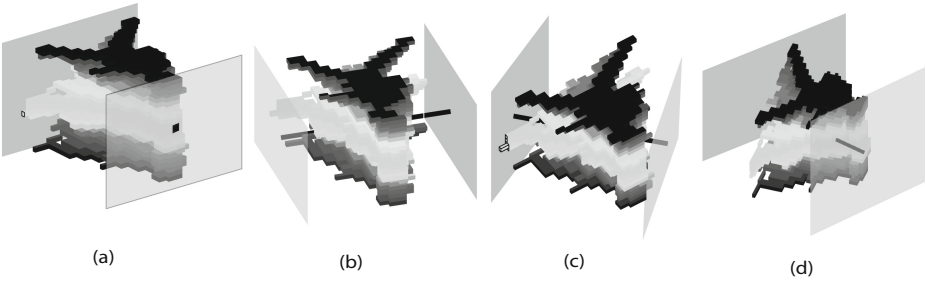


Fig. 6. View directions used in the experiments.

Table 2. Correct classification rates for gait based recognition.

Method	0	18	36	54	72	90	108	126	144	162	180	**Avg**
Wavelet(FD) [1]	100	100	100	93.4	81.1	90.3	90.3	83.3	91.9	92.7	97.6	92.9
Our method	99.3	99.1	98.8	98.3	97.6	98.0	98.3	98.3	98.2	98.2	99.0	98.5

normally. All the sequence was selected as the subsequence of representation, and the n parameter was set to 24.

Only lateral view was used in this case. For each person walking normally the 6 sequences provided by the database were used, while for parsons carrying bag the 2 sequences provided by the database were used. The same 4 view directions used for gait based recognition were used in this case.

It is important to point out that different kinds of bags carried by the persons, as well as the variation of position where they carry those bags, makes harder the classification, in (Fig. 7) are shown some details. Another hard situation emerges when the body occludes the bag, in this case even humans show difficulties to detect the bag.

Fig. 7. CAISA-B carrying bag images.

In our experimentation protocol, the 200 subjects to analyze were divided in 100 disjoint sets, each containing a person carrying a bag and a person without bag. One of these sets was used for testing, while the remaining 99 were used for training. The average of the 100 results obtained from the cross validation gives

up a correct classification rate of 92.5 % in the case of persons carrying bags, and 95.1 % in the case of persons walking in natural conditions.

3.4 Simple Action Recognition

In the case of simple action recognition, two action were taken from the KTH database, namely the action to run and the action to walk. Usually, to descry a person running gives the idea of the occurrence of an unusual event, which argues the importance of detecting this action in real surveillance scenarios.

Experiment 4. In order to test the performance of the proposed method, 20 persons were selected from the 25 persons in KTH database, 10 of them running and 10 walking. The subsequence of representation consists of all the sequence, and only lateral view was used.

The same four view directions used for gait recognition were selected in this case, while the n parameter took 24 as value. For each person four video sequences were used to carry out the experiment. The videos were filmed in an outdoor scenario as shown in (Fig. 8).

Fig. 8. Persons running and walking taken from the KTH database.

The 20 subjects selected were partitioned in 10 disjoint sets, each containing 2 subjects, one running and one walking. Only one of these 10 sets was used for testing, and the remaining 9 sets were used for training. The correct classification rate is the average of the 10 results obtained from the cross validation. In the case of persons running the result was 97.7 %, while for persons walking the result was 97.5 %.

4 Conclusions and Future Work

This paper shows that is possible to use the same topological feature, previously used for gait based human identification at a distance, in other tasks concerning to activities monitoring, including gender classification, carrying bag detection and simple action recognition. This kind of features have showed to be robust and discriminant. There are many issues to take into account in future works

in order to improve the promising results obtained in this work. Also, in future works, it is possible to model the crowd and its behavior as a whole and interpret the movements of the different parts characterizing the dynamic of holes and connected components.

References

1. Chen, C.H., Liang, J.M., Zhao, H., Hu, H.H., Tian, J.: Frame difference energy image for gait recognition with incomplete silhouettes. PRL **30**(11), 977–984 (2009)
2. Hilaga, M., Shinagawa, Y., Kohmura, T., Kunii, T.L.: Topology matching for fully automatic similarity estimation of 3d shapes. In: Proceedings of the 28th Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '01, pp. 203–212 (2001)
3. Bronstein, A.M., Bronstein, M.M., Kimmel, R.: Topology invariant similarity of nonrigid shapes. Int. J. Comput. Vis. **81**(3), 281–301 (2009)
4. Zomorodian, A., Carlsson, G.: Localized homology. Comput. Geom. **41**(3), 126–148 (2008)
5. Zomorodian, A.: Computational topology. In: Atallah, M., Blanton, M. (eds.) Algorithms and Theory of Computation Handbook, vol. 2, 2nd edn. Chapman & Hall/CRC Press, Boca Raton (2010)
6. Zomorodian, A.: Topology for Computing. Cambridge University Press, New York (2009)
7. Ghrist, R.: Barcodes, the persistent topology of data. BAMS. Bull. Am. Math. Soc. **45**, 61–75 (2008)
8. Lamar-León, J., García-Reyes, E.B., González-Díaz, R.: Human gait identification using persistent homology. In: Álvarez, L., Mejail, M., Gómez, L., Jacobo, J. (eds.) CIARP 2012. LNCS, vol. 7441, pp. 244–251. Springer, Heidelberg (2012)
9. Lamar, J., Garcia, E., Gonzalez-Diaz, R., Alonso, R.: An application for gait recognition using persistent homology. Electron. J. Image-A **3**(5) (2013)
10. Nixon, M.S., Carter, J.N.: Automatic recognition by gait. Proc. IEEE **94**(11), 2013–2024 (2006)
11. Murray, M.P.: Gait as a total pattern of movement: including a bibliography on gait. Am. J. Phys. Med. Rehabil. **46**(1), 290–333 (1967)
12. Winter, D.A.: Biomechanics and Motor Control of Human Gait: Normal, Elderly and Pathological. University of Waterloo Press, Waterloo (1991)
13. Golomb, B.A., Lawrence, D.T., Sejnowksi, T.J.: SEXNET: a neural network identifies sex from human faces. In: Lippmann, R.P., Moody, J.E., Touretzky, D.S. (eds.) Advances in Neural Information Processing Systems, vol. 3, pp. 572–579. Morgan Kaufmann Publishers Inc., San Mateo (1991)
14. Harb, H., Chen, L.: Gender identification using a general audio classifier. In: Proceedings of the 2003 International Conference on Multimedia and Expo, ICME '03, vol. 1, pp. 733–736. IEEE (2003)
15. Yu, S., Tan, T., Huang, K., Jia, K., Wu, X.: A study on gait-based gender classification. IEEE Trans. Image Process. **18**(8), 1905–1910 (2009)
16. Hu, M., Wang, Y., Zhang, Z., Zhang, D.: Gait-based gender classification using mixed conditional random field. IEEE Trans. Syst. Man Cybern. Part B **41**(5), 1429–1439 (2011)
17. Kale, A., Sundaresan, A., Rajagopalan, A.N., Cuntoor, N.P., Chowdhury, A.K.R., Kruger, V., Chellappa, R.: Identification of humans using gait. IEEE Trans. Image Process. **13**(9), 1163–1173 (2004)

18. Goffredo, M., Carter, J.N., Nixon, M.S.: Front-view gait recognition. In: Biometrics: Theory, Applications, and Systems, pp. 1–6, 29 September–1 October (2008)
19. Aggarwal, J.K., Cai, Q.: Human motion analysis: a review. Comput. Vis. Image Underst. **73**(3), 428–440 (1999)
20. Mather, G., Murdoch, L.: Gender discrimination in biological motion displays based on dynamic cues. Proc. Biol. Sci. **258**(1353), 273–279 (1994)
21. Dondera, R., Morariu, V.I., Davis, L.S.: Learning to detect carried objects with minimal supervision. In: CVPR Workshops, pp. 759–766. IEEE (2013)
22. Senst, T., Evangelio, R.H., Eiselein, V., Pätzold, M., Sikora, T.: Towards detecting people carrying objects - a periodicity dependency pattern approach. In: Richard, P., Braz, J. (eds.) VISAPP (2), pp. 524–529. INSTICC Press (2010)
23. BenAbdelkader, C., Davis, L.: Detection of people carrying objects: A motion-based recognition approach. In: Proceedings of Fifth IEEE International Conference on Automatic Face and Gesture Recognition, pp. 378–383. IEEE (2002)

TLD and Struck: A Feature Descriptors Comparative Study

Francesco Adamo[1(✉)], Pierluigi Carcagnì[1], Pier Luigi Mazzeo[2], Cosimo Distante[2], and Paolo Spagnolo[2]

[1] Faculty of Engineering, University of Salento, Lecce, Italy
francesco.adamo@unisalento.it
[2] National Research Council of Italy - Institute of Optics, Lecce, Italy

Abstract. Object tracking across multiple cameras is a very challenge issue in vision based monitoring applications. The selection of features is the first step to realize a reliable tracking algorithm.

In this work we analyse TLD and Struck, which are two of the most cited real-time visual trackers proposed in the literature in last years. They use two different feature extraction methodologies, Fern and Haar, respectively. The idea of this work is to compare performance of these well known visual tracking algorithms replacing their original feature characterization methods with local feature-based visual representations.

We test the improvement in terms of object detection and tracking performance grafting different features characterization into two completely different online tracker frameworks.

The used feature extraction methods are based on Local Binary Pattern (LBP), Local Gradient Pattern (LGP) and Histogram of Oriented Gradients (HOG). LGP is a novel detection methodology which is insensitive to global intensity variations like other representations such as local binary patterns (LBP).

The experimental results on well known benchmark sequences show as the feature extraction replacing improve the overall performances of the considered real-time visual trackers.

1 Introduction

Automatic object tracking [29] is one of the most important application in computer vision (surveillance, human robot interaction and medical imaging).

Moreover, in many applications, it is important to locate and track targets, maintaining their identities when they travel within or across different cameras.

The first aim of the visual tracking is to estimate the state of the target, in a frame, along the execution of video sequence.

There are many factors that affect the efficiency of a tracking algorithm, such as the variation of illumination, the presence of occlusions and the distinction of the object from the background [26].

If multiple cameras are considered then the inter-camera association problem must be solved: the appearance of a target in different cameras may not be

P.L. Mazzeo et al. (Eds.): AMMDS 2014, pp. 52–63, 2014.
DOI: 10.1007/978-3-319-13323-2_5

consistent (different sensor characteristics, lighting conditions, viewpoints, ...) and the information of tracked objects between cameras becomes much less reliable.

However, the problem of feature selection is common to both single and multi view tracking approaches [16].

Two of the most cited real-time trackers proposed in the literature in recent years are TLD and Struck. TLD (Tracking - Learning - Detection) [13] is an architecture for long-term tracking of unknown objects in a video sequence. This tracker decomposes the long-term tracking task into three subtasks: tracking, learning and detection. The tracker follows the object from frame to frame. The detector localizes all appearances that have been observed so far and corrects the tracker if necessary. The learning estimates detector's errors and updates it to avoid these errors in the future. TLD can work in real-time on PC equipped with standard hardware.

Struck (Structured Output Tracking with Kernels) [9] is a framework for adaptive visual object tracking that is based on structured output prediction. This framework uses a kernelized structured output support vector machine (SVM), which is learned on-line to provide adaptive tracking.

The application can work in real-time thanks to a budgeting mechanism which prevents the unbounded growth in the number of support vectors.

In both architectures, challenging problems, like the similarity appearance measures and object changes appearance managing, remain still open.

Often, the appearance of different objects and background in the scenes may be similar to the target appearance and this may influence its correct detection. This way, it is difficult to distinguish the features of the target object from those of the objects in the scene that are not targets. This phenomenon is known as cluttering [14].

For this reasons feature representation is a critical aspect in visual tracking.

In TLD the features are characterized by the random Fern [18], while in Struck simple Haarlike features [25] are used. In this work, we replace these feature characterization (Fern and HaarLike) with different local feature-based method, such as LBP (Local Binary Pattern) [17], LGP (Local Gradient Patterns) [11] and HOG (Histogram of Oriented Gradients) [4].

LBP method has been intensively used as features extractor in visual tracking: in [3] it is used for face description in order to achieve robust face tracking performance.

The face is represented by the fusion of color and LBP cue. However, because only LBP cue is employed, the tracking method is not robust. In [28] a novel on-line feature selection mechanism based on mean shift tracking algorithm is described. Different features as gray level, Local Binary Pattern (LBP) and edge orientation are selected, but they are tested only with mean shift tracking algorithm.

An LBP-based tracking algorithm is described in [19]. This approach is based on two main steps: (i) LBP histogram of each image and target pattern are constructed; (ii) a similarity measure is calculated in order to find the best LBP-histograms matching.

The idea is interesting because LBP is used to solve visual tracking; however, no tests have been done on tracking video benchmark. LBP feature, together with its variants, have been widely explored and used to solve classification problems in different contexts, mainly in biometric analysis: face detection [30], face recognition [1], facial expression recognition [20], gender analysis [22].

In [2] several methods of pedestrian detection that use different local statistical measures such as uniform local binary patterns (LBP) and a modified version of histogram of oriented gradients (HOGs) are presented. Two extraction features methods are compared, but the results are presented only in the particular context of pedestrian detection.

Some HOG based tracking systems have been presented in literature: in [23] is proposed a real-time visual tracking system that delivers high performance under difficult situations. The system is based on Histogram of Oriented Gradient (HOG) within the on-line boosting framework.

The comparison, however, is done only with Haar-based tracking system.

In [27] a detection and tracking algorithm for pedestrian is presented. It is based on HOG and Support Vector Machine (SVM) as detector and particle filtering as tracker. Experiments show that using HOG as features give better pedestrian detection results. Recently, the HOG feature has been widely used in several applicative contexts, including human [6] and object [8] detection.

The literature does not present any work in which the novel LGP representation has been used for visual tracking. LGP representation is insensitive to both global intensity variations like other representations such as local binary patterns (LBP) and local intensity variations along the edge components. It well suited as feature descriptors for visual tracking problem. For the first time, we employ the LGP in visual tracking comparing its effect in two different tracking frameworks.

The multi-camera tracking problem has been addressed, instead, in several articles, such as [16] and [15]. In particular, in [16] is proposed a novel target re-identification method in order to estimate people movements in non-observed regions between camera views. The method is based on a modification of the Social Force Model (SFM) and takes into account barrier avoidance constraints as well as the desired motion toward specific goals in the scene. In [15] authors propose a novel system for associating multi-target tracks across multiple non-overlapping cameras by an on-line learned discriminative appearance affinity model. The main contribution of this paper is focused on learning a discriminative appearance affinity model at runtime. In order to solve the ambiguous labelling problem, a Multiple Instance Learning (MIL) is applied to learn an appearance affinity model which effectively combines three complementary image descriptors and their corresponding similarity measurements.

In [21] a robust hand tracking approach for unconstrained videos based on modified Tracking - Learning - Detection (TLD) algorithm is presented. In this work the authors introduce a back projection algorithm applied to the detection phase of the TLD algorithm in order to improve the tracking rate.

Generally speaking, in this paper we investigate on the feasibility to replace TLD and Struck feature descriptors in order to improve the overall detection and tracking performances of these both very popular algorithms.

This work offers a starting point for solution to re-identification problem, through the use of a better features characterization. Moreover, the algorithm is only developed for single-camera scenario: we assume that is possible to re-identify an object in a scene different from that in which it was identified the first time, if the characterization of its features is robust.

The performances are evaluated on the most important video benchmark sequences[1]. The obtained results confirm that replacing the feature extraction mechanism gives better results in the both algorithm architectures with respect to the classical ones. The paper is organized as follows.

In Sect. 2 there is a brief explanation of the feature extraction methods actually used in the two considered frameworks. The considered methods are briefly presented.

Experimental results are described and discussed in Sect. 3.

Conclusive remarks and future perspectives are given in Sect. 4.

2 Visual Feature Representation

TLD uses a method named random Fern classification.

This method is based on non-hierarchical structures (fern) which are able to classify patches. Each one consists of a small set of binary tests and returns the probability that a patch belongs to any one of the classes that have been learned during training. These responses are then combined in a Naive Bayesian classifier.

In Struck is straightforward to use different image features by modifying the kernel function used for evaluating patch similarity. Main results are obtained using simple Haar-like features for image representation.

Fern and Haar descriptors have been replaced by Local Binary Patterns (LBP), Local Gradient Patterns (LGP) and Histogram of Oriented Gradients (HOG).

In the following subsection these descriptors are briefly described.

2.1 Local Binary Patterns

The LBP operator proposed in [17] is a powerful tool for describing the texture of images. The original LBP operator labels the pixels of an image by thresholding the 3×3 - neighbours of each pixel with the center value and considering the result as a binary number. Then the histogram of the labels is used as a texture descriptor.

In Fig. 1 an example of it is proposed. Each image point in the 3×3 size is compared with the central value and thresholded, producing a 3×3 binary

[1] http://www.cvg.rdg.ac.uk/PETS2009

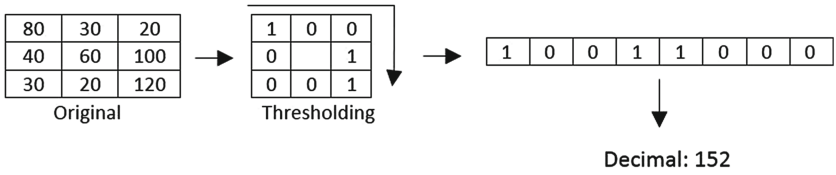


Fig. 1. The original LBP operator

matrix. Then, this matrix is vectorialized by considering a predefined pattern on it (in this example, the starting point is the pixel in the first position, and the matrix is clockwise read). The resulting vector is considered as the 8-bit representation of a decimal number. The main limitation of the basic LBP operator is its fixed small spatial area which can not capture the larger texture. So the LBP operator is extended to use different sizes [24]. Bi-linearly interpolating the pixel values and the use of circular neighbours make any scale and number of pixels in the neighbours possible.

2.2 Local Gradient Patterns

In [11] is presented a novel features representation method called LGP (Local Gradient Patterns), which generates constant patterns irrespective of local intensity variations along edges.

The Local Gradient Pattern operator computes, like as LBP operator, a binary code for a given pixel. The LGP operator uses the gradient values of the eight neighbours of a given pixel: the absolute value of the intensity difference between the given pixel and its neighbouring ones is computed. Then, the average of the gradient values of the eight neighbouring pixels is computed and assigned at the given pixel. This value is used as threshold to compare the value of the eight neighbouring pixels. If the gradient value of a neighbouring pixel is greater than the threshold value, a value of 1 is assigned, otherwise a value of 0 is assigned. Then, the concatenation of the binary ones and zeros into a binary code produces the LGP code for a given pixel (Fig. 2).

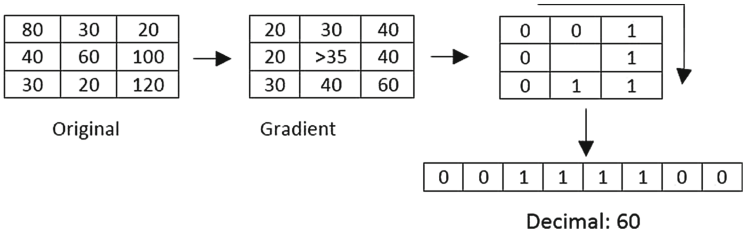


Fig. 2. The original LGP operator

As the LBP, the LGP operator can be extended to use different-sized neighbours. A circle with a prefixed radius and centred on a specified pixel is considered; the sampling points lie on the circle (Fig. 3).

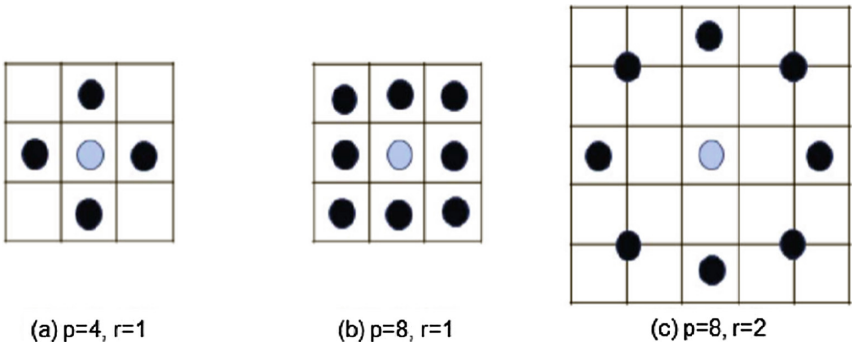


Fig. 3. Three examples of neighbouring pixels: $LGP_{1,4}$, $LGP_{1,8}$, $LGP_{2,8}$, $LBP_{1,4}$, $LBP_{1,8}$, $LBP_{2,8}$

In [11] a good overview on how LBP and LGP generate the similar and different codes depending on global and local intensity changes is proposed.

2.3 Histogram of Oriented Gradients

Histogram of Oriented Gradients (HOG) [4], is a feature descriptor: this technique counts occurrences of gradient orientation in localized portions of an image.

This method is based on evaluating well-normalized local histograms of image gradient orientations in a dense grid. Local object appearance and shape can often be characterized fairly well by the distribution of local intensity gradients or edge directions, even without precise knowledge of the corresponding gradient or edge positions. The implementation requires that the image window is divided into small spatial regions, called cells, and for each cell is accumulated a local 1-D histogram of gradient directions or edge orientations over the pixels of the cell. The combined histogram entries form the representation.

For better invariance to disturb, it is also useful to contrast-normalize the local responses before using them. This can be done by accumulating a measure of local histogram energy over somewhat larger spatial regions, named blocks, and using the results to normalize all of the cells in the block. The normalized descriptor blocks represent the Histogram of Oriented Gradient (HOG) descriptors.

3 Experiments

In order to compare several features extraction methods, some changes have been made to TLD and Struck framework.

In fact, the original systems of features characterization are replaced with the different methods, such as LBP, LGP and HOG.

For LBP we used an available implementation on-line[2].
The VLFeat library[3] is used for HOG operator.
Instead, a freely available LGP implementation is available by the author[4].

3.1 Video Sequences

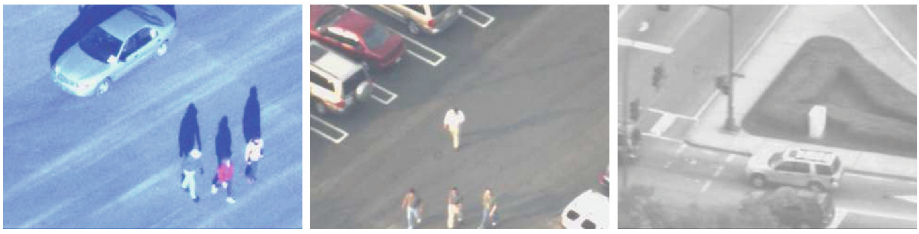
The following six sequences, taken from PETS 2009 (see Fig. 4), were used for evaluating object tracking methods. The sequences that are used for evaluation, are all accompanied by manually annotated ground truth data. The sequence "David Indoor" consists of 761 frames and shows a person walking from an initially dark setting into a bright room. In this sequence there are not occlusions.

The sequence "Jumping" consists of 313 frames and shows a person jumping rope, which causes motion blur.

The sequences named "Pedestrian 1" (140 frames), "Pedestrian 2" (338 frames) and "Pedestrian 3" (184 frames) show pedestrians being filmed by an unstable camera.

The sequence Car consists of 945 frames showing a moving car. This sequence exhibits low contrast and various occlusions occur.

(a) david indoor (b) jumping (c) pedestrian1



(d) pedestrian2 (e) pedestrian3 (f) car

Fig. 4. PETS2009

[2] http://www.bytefish.de/blog/local_binary_patterns/
[3] http://www.vlfeat.org
[4] For LGP source code: pierluigi.carcagni@ino.it.

3.2 Overlap Measure

The overlap measure in Eq. 1 is used to compare the output of an algorithm to ground truth values. This measure equally penalizes translations in both directions and scale changes as is shown in [10].

The overlap formula is:

$$overlap = \frac{B1 \cap B2}{B1 \cup B2} = \frac{I}{B1 + B2 - I} \tag{1}$$

where *B1* is the area of the first bounding box, *B2* the area of the second bounding box and I is the area of the intersection of the two bounding boxes.

A result is considered true positive (TP) if the overlap is larger than a threshold ω. A result is counted as false negative (FN) when the algorithm does not produce some results for a frame even though there is an entry in the ground truth database. In the opposite case is counted a false positive (FP). If for a frame neither an algorithmic output nor an entry in the ground truth database exist then this case is considered a true negative (TN). In the end, both a false negative (FN) and a false positive (FP) are counted if the overlap is lower than the threshold ω.

After processing, for each video sequence the two parameters *recall* and *precision* are calculated:

$$recall = \frac{TP}{TP + FN} \tag{2}$$

$$precision = \frac{TP}{TP + FP}. \tag{3}$$

Equation 2 measures the fraction of positive examples that are correctly labeled. Instead, Eq. 3 measures the fraction of examples classified as positive that are truly positive [5].

3.3 Experimental Results

The performances of algorithmic output depend on the threshold ω that defines the bounding box overlap between the algorithmic results and ground truth values.

Recall and precision of each video sequence are calculated for three threshold (ω) values: 0.75, 0.50 and 0.25.

When ω decreases both recall and precision increase.

Of course, in the Tables 1 and 2 we report the results, obtained for $\omega = 0.25$. The results are referred only to original LBP and LGP operators.

In Table 1 and in Table 2 the obtained results are compared with the results presented in original TLD [12] and with the results obtained running Struck[5]. The maximum recall and precision values for a sequence are bolded.

[5] http://www.samhare.net/research/struck

Table 1. Comparison between obtained results and original frameworks results (TLD).

Comparison	*LBP*	*LGP*	*HOG*	*TLD* [12]
David indoor	**0.98/0.99**	**0.98/0.99**	**0.98/0.99**	0.94/0.94
Jumping	0.53/1	**0.96/1**	0.95/1	0.77/0.86
Pedestrian1	0.01/1	0.01/1	0.01/1	**0.16/0.22**
Pedestrian2	0.20/1	0.20/1	0.20/1	**0.95/1**
Pedestrian3	**0.94/1**	0.93/1	0.92/1	**0.94/1**
Car	**0.96/1**	**0.96/1**	0.78/0.84	0.83/0.93

Table 2. Comparison between obtained results and original frameworks results (Struck).

Comparison	*LBP*	*LGP*	*HOG*	*Struck*
David indoor	**1/1**	0.97/0.97	**1/1**	0.17/0.17
Jumping	**1/1**	0.64/0.64	**1/1**	**1/1**
Pedestrian1	**0.97/0.97**	0.69/0.69	0.66/0.66	0.86/0.86
Pedestrian2	**0.91/0.72**	0.65/0.51	0.39/0.30	0.38/0.30
Pedestrian3	**1/0.85**	**1/0.85**	0.95/0.80	**1/0.85**
Car	0.84/0.76	0.92/0.84	0.88/0.80	**0.93/0.84**

Table 1 shows as the proposed methods for features extraction, generally improve the traditional TLD implementation.

In particular, the greater improvement is on the *"Jumping"* and *"David Indoor"* sequences because LGP and HOG (based on gradient) retain the patch texture features and are robust with illumination changes.

It should be noted that in the case of *"Pedestrian(1,2)"* traditional TLD conserves the best results.

This is normal because the pedestrian patches do not have a discriminative texture.

In *"Pedestrian(3)"* sequence, instead, the good results are due to absence of shadows.

"Car" sequence presents for LBP, LGP and HOG similar results. They work better than Fern because the features are more discriminative.

Anyway, when the object we want to track does not have distinguishable high frequency components, our local feature method could fail.

In Table 2 the Struck framework performances are compared.

LBP performs better almost in all sequences (except only *"car"*). HOG outperforms all other descriptors only in *"David indoor"* and *"Jumping"* (same performances LBP) where patches to be tracked are not surrounded by shadows. In sequence *"Pedestrian3"* HOG gives the worst results (respect the LBP, LGP and Haar) because of the poor texture content of the tracked patch. The best

results for *"car"* sequence, are obtained by using Haar descriptors. It's to be noted that these results are very close to those obtained with LGP and HOG detector. This is because the Haar feature better contains the low frequencies of the tracked patch.

Because the aim of this work is to compare the discriminative power of the different local-based feature descriptors, we do not measure the computational load. Anyway, even if the time consumed for different local feature calculation is grown, the computational performances remain close to the real-time (Intel I7 processor based system).

4 Conclusions

In this work we have investigated on the feasibility of improving performances on visual tracking algorithms: TLD and Struck. The original feature characterization methods have been replaced with local feature-based visual representations. For the first time, we have used the LGP operator in visual tracking algorithms. Furthermore LBP and HOG descriptors have been used in TLD and Struck architectures. The preliminary results show that some improvements are possible. LBP in Struck architecture works better than Haar in all the tested sequences. However, we evaluated the possibility to use features extracted from images captured by a given camera and to compare these with the features acquired by a different camera, in order to retrieve object target.

Future works will be addressed to study different classification algorithms which are more suitable for each feature characterization. Moreover, we are testing the proposed approach on a public multi-view dataset [7].

Acknowledgment. This work has been supported by the "2007–2013 NOP for Research and Competitiveness for the Convergence Regions (Calabria, Campania, Puglia and Sicilia)" with code PON04a3_00201 and in part by the PON Baitah, with code PON01_980.

References

1. Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: application to face recognition. IEEE Trans. Pattern Anal. Mach. Intell. **28**(12), 2037–2041 (2006)
2. Brehar, R., Nedevschi, S.: Local information statistics of LBP and HOG for pedestrian detection. In: 2013 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP), pp. 117–122 (2013)
3. Chuan-xu, W., Zuo-yong, L.: A new face tracking algorithm based on local binary pattern and skin color information. In: International Symposium on Computer Science and Computational Technology 2008, ISCSCT '08, vol. 2, pp. 657–660 (2008)
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2005, CVPR 2005, vol. 1, pp. 886–893. IEEE (2005)

5. Davis, J., Goadrich, M.: The relationship between precision-recall and roc curves. In: Proceedings of the 23rd International Conference on Machine Learning, pp. 233–240. ACM (2006)
6. Dollar, P., Belongie, S., Perona, P.: The fastest pedestrian detector in the west. In: Proceedings of BMVC, pp. 68.1–68.11 (2010). doi:10.5244/C.24.68
7. D'Orazio, T., Leo, M., Mosca, N., Spagnolo, P., Mazzeo, P.L.: A semi-automatic system for ground truth generation of soccer video sequences. In: Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance 2009, AVSS'09, pp. 559–564. IEEE (2009)
8. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Trans. Pattern Anal. Mach. Intell. **32**(9), 1627–1645 (2010)
9. Hare, S., Saffari, A., Torr, P.H.: Struck: structured output tracking with kernels. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 263–270. IEEE (2011)
10. Hemery, B., Laurent, H., Rosenberger, C.: Comparative study of metrics for evaluation of object localisation by bounding boxes. In: Fourth International Conference on Image and Graphics 2007, ICIG 2007, pp. 459–464 (2007)
11. Jun, B., Choi, I., Kim, D.: Local transform features and hybridization for accurate face and human detection. IEEE Trans. Pattern Anal. Mach. Intell. **35**(6), 1423–1436 (2013)
12. Kalal, Z., Matas, J., Mikolajczyk, K.: P-N learning: Bootstrapping binary classifiers by structural constraints. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 49–56. IEEE (2010)
13. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. IEEE Trans. Pattern Anal. Machine Intell. **34**(7), 1409–1422 (2012)
14. Khan, Z., Gu, I.H., Backhouse, A.: Robust visual object tracking using multi-mode anisotropic mean shift and particle filters. IEEE Trans. Circuits Syst. Video Technol. **21**(1), 74–87 (2011)
15. Kuo, C.-H., Huang, C., Nevatia, R.: Inter-camera association of multi-target tracks by on-line learned appearance affinity models. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 383–396. Springer, Heidelberg (2010)
16. Mazzon, R., Cavallaro, A.: Multi-camera tracking using a multi-goal social force model. Neurocomputing **100**, 41–50 (2013)
17. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. Pattern Recogn. **29**(1), 51–59 (1996)
18. Ozuysal, M., Fua, P., Lepetit, V.: Fast keypoint recognition in ten lines of code. In: IEEE Conference on Computer Vision and Pattern Recognition 2007, CVPR'07, pp. 1–8. IEEE (2007)
19. Rami, H., Hamri, M., Masmoudi, L.: Article: objects tracking in images sequence using local binary pattern (LBP). Int. J. Comput. Appl. **63**(20), 19–23 (2013). (Published by Foundation of Computer Science, New York, USA)
20. Shan, C., Gong, S., McOwan, P.W.: Facial expression recognition based on local binary patterns: a comprehensive study. Image Vis. Comput. **27**(6), 803–816 (2009)
21. Shi, H., Lin, Z., Tang, W., Liao, B., Wang, J., Zheng, L.: A robust hand tracking approach based on modified tracking-learning-detection algorithm. In: Park, J.J.J.H., Chen, S.-C., Gil, J.-M., Yen, N.Y. (eds.) Multimedia and Ubiquitous Engineering. LNEE, vol. 308, pp. 9–15. Springer, Heidelberg (2014). http://dx.doi.org/10.1007/978-3-642-54900-7_2

22. Sun, N., Zheng, W., Sun, C., Zou, C., Zhao, L.: Gender classification based on boosting local binary pattern. In: Wang, J., Yi, Z., Żurada, J.M., Lu, B.-L., Yin, H. (eds.) ISNN 2006. LNCS, vol. 3972, pp. 194–201. Springer, Heidelberg (2006)
23. Sun, S., Guo, Q., Dong, F., Lei, B.: On-line boosting based real-time tracking with efficient hog. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2297–2301 (2013)
24. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Anal. Mach. Intell. **24**(7), 971–987 (2002)
25. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2001, CVPR 2001, vol. 1, pp. I-511. IEEE (2001)
26. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: a benchmark. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2411–2418. IEEE (2013)
27. Xu, F., Gao, M.: Human detection and tracking based on hog and particle filter. In: 2010 3rd International Congress on Image and Signal Processing (CISP), vol. 3, pp. 1503–1507 (2010)
28. Yi, S., Yao, Z., Liu, J., Chen, J., Liu, W.: Robust tracking using on-line selection of multiple features. In: 2012 Spring Congress on Engineering and Technology (S-CET), pp. 1–5 (2012)
29. Yilmaz, A., Javed, O., Shah, M.: Object tracking: a survey. ACM Comput. Surv. (CSUR) **38**(4), 13 (2006)
30. Zhang, L., Chu, R.F., Xiang, S., Liao, S.C., Li, S.Z.: Face detection based on multi-block LBP representation. In: Lee, S.-W., Li, S.Z. (eds.) ICB 2007. LNCS, vol. 4642, pp. 11–18. Springer, Heidelberg (2007)

Group Sleepiness Measurement in Classroom

Kenshiro Nishikawa(✉) and Mineichi Kudo

Hokkaido University, Sapporo, Japan
{kenshiro,mine}@main.ist.hokudai.ac.jp

Abstract. We propose a simple method for measuring group sleepiness, aiming at making clear how the environmental factor affects such sleepiness. A unit to measure the angle of head slant and the acceleration of head moving was attached to one side of ears with a bandana. A video interface was also developed to display how many students in a classroom are awake, felling sleepiness, or sleeping with the corresponding colors. The experiments showed the existence of some different patterns in the way of falling asleep. We revealed that level prediction of group sleepiness is easier than that of individual sleepiness. Indeed, the standard error was reduced from 0.695 (for individual) into 0.423 (for group), showing almost perfect prediction of sleepiness level of a group of five persons.

1 Introduction

Sleep is necessary and important for us to live healthy and work regularly. However, it can be harmful or even dangerous if one feels sleepiness in unexpected situations such as driving. Especially it is dangerous when it happens at a factory during executing a hard task requiring strong concentration. The factor and timing of feeling drowsy would be different from person to person and their physical conditions as well. However, in a situation where an environmental factor is most influential to the drowsiness, many workers/persons sharing the same environment would feel sleepiness at the same time, such as students listening to a lecture, especially when it is delivered in a monotonic voice in a warm afternoon. This is a study to know when and to what degrees a group feels drowsy.

Many studies on sleepiness have been made (for example, see [1,2]). Most studies among them measure the brain wave or eye movement of the subject. The electroencephalogram (EEG) is available for predicting sleepiness with a high accuracy. However, since EEG is measured by skin contact-type sensors, the practical use is limited. On the contrary, the eye movement is easily analyzed by a video camera, but it assumes the subject to hold his/her posture still.

In any case of those studies, the target/subject is one person. That is, their goal is to detect a single person's sleepiness and to evaluate the degree as well. On the contrary, this study aims at detecting a group's sleepiness with the degree. Only a few studies deal with group sleepiness [3]. In [3], the authors try to find out sleeping students in a classroom using a video camera. However, camera images can be easily occluded by some obstacles and are weak for the change of illumination.

P.L. Mazzeo et al. (Eds.): AMMDS 2014, pp. 64–72, 2014.
DOI: 10.1007/978-3-319-13323-2_6

2 Equipment

We developed a sensor device with a wireless transmitter for measuring head acceleration (Fig. 1). The sensor is 3-axis acceleration sensor (KXM52-1050, Kionix lnc. [4]). The wireless module is XBee (DIGI-XB24-Z7WIT-004, Digi international [5]). The specification of the device is shown in Table 1. The appearance is shown in Figs. 1 and 2. The device is light and small enough for usage like Fig. 2. In the following experiments, this unit is attached to one side of subject's ears (Fig. 2). The X-axis of acceleration sensor is positive for frontward of the body, Y-axis is for sideward in the left hand side, Z-axis is for vertically downward. We use the angle θ mainly (Fig. 3). The hardware construction is shown in Fig. 4.

Fig. 1. Device for measuring individual sleepiness

Fig. 2. Device attached to the ear side with a bandana

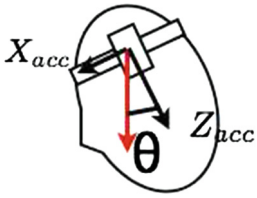


Fig. 3. Angle of head slant. It is calculated from the acceleration values in direction X and gravity.

3 Methodology

We measured the change of angle θ (Fig. 3) of head slant over time. In a period of thirty seconds (150 points in sampling at 5 Hz), we calculated the mean and variance of θ's. There are four levels of sleepiness; 0: awake, 1: light drowsy, 2: drowsy, 3: sleeping, as defined in Table 2. In addition, we defined the group sleepiness as the mean of individual sleepiness.

Table 1. Specification of device

Acc sensor	
Maximum load	±2G
Sensitivity	660 mV/g
Bandwidth	0 to 1500 Hz
Operating Temperature	−40 to 85°
Power supply	3.3 V
Wirelessmodule	
Indoor/Urban Range	30 m
Maximum association number	65536
RF data rate	250,000 bps
Unit	
Dimension	$40 \times 55 \times 20\,\mathrm{mm}^2$
Weight (unit)	15 g
Weight (power source)	70 g

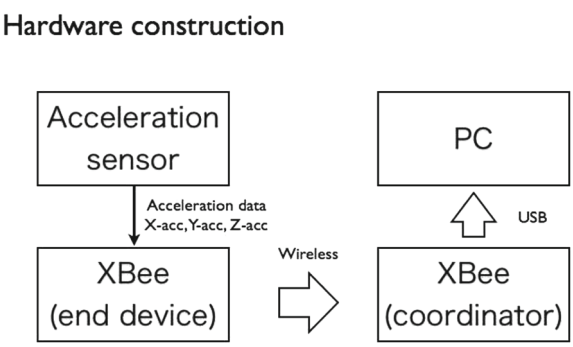


Fig. 4. Hardware construction in the device

3.1 Measurement of the Angle of Head Slant

The angle of head slant was measured by an acceleration sensor attached to one side of ears with a bandanna (Fig. 2). It tells three values of X, Y, Z axes. To remove noisy acceleration, a low-pass filter was applied. The gravity acceleration (calculated from three axes values) are safely obtained. The calibration is carried out to determine the initial head position at the starting time of experiments.

3.2 Rule for Prediction

We adopted a simple rule for predicting the level of sleepiness from a short-term sequence of head angles. To do this, we used the mean μ and the variance σ^2 of head angle θ's for recent 150 points obtained the current time. The rule for

Table 2. Definition of sleepiness level

Sleepiness level	State	Behavior
0 (shallowest)	Awake	Facing forward
1	Light drowsy	Lowering one's eyes
2	Drowsy	Nodding
3 (deepest)	Sleep	Facing down and staying still

Prediction Algorithm

Calculate μ, σ from θ for recent 30 seconds
if $\mu > \theta_1$ **then**
  sleepiness level 0
else if $\mu < \theta_2$ **then**
  sleepiness level 3
else if $\sigma < \rho$ **then**
  sleepiness level 1
else
  sleepiness level 2

Fig. 5. Rule for prediction. Here μ is the mean angle of the head slant and σ is the standard deviation.

prediction is shown in Fig. 5. With three thresholds θ_1, θ_2 and μ, we classify the state into one of three levels of 0 (awake), 3 (sleep) and the other (1 (lightly drowsy) or 2 (drowsy)).

4 Experiment on Individual Sleepiness

To evaluate the basic performance of our device for measuring individual sleepiness, we carried out a simple experiment. Five subjects are asked to perform a sequence of behaviors each of which corresponds to 0 of 3 levels of sleepiness.

The given instruction was as follows : face forward for 2 min, lower eyes for 1 min, nod for 1 min and sleep for 3 min. They performed it separately. The measured change of angles of head slant is shown in Fig. 6. As seen in Fig. 6, there are a variety of difference in the way of falling into sleep. The period of nodding is different over subjects and the angle of sleeping posture is different as well. The most difficult task is to distinguish between level 1 (light drowsy) and level 2 (drowsy).

We determined the values of parameters in the rule for prediction of sleepiness level from these graphs of five subjects. These values were set to $\theta_1 = -15$, $\theta_2 = -40$ and $\rho = 10$. Using these values, their actual sleepiness level can be predicted well as shown in Fig. 7.

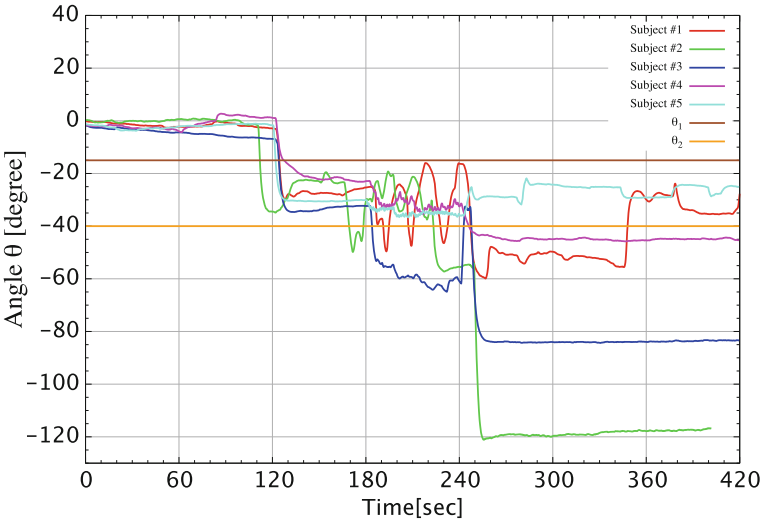


Fig. 6. Individual sleepiness. Five subjects were instructed to pretend to sleep at the specified level (0–3). The graphs are their head angles. Using thresholds θ_1, θ_2 and ρ for the short-term mean and for the standard deviation of head angles, their sleepiness level is predicted (the result is not shown) ($\theta_1 = -15$, $\theta_2 = -40$, $\rho = 10$).

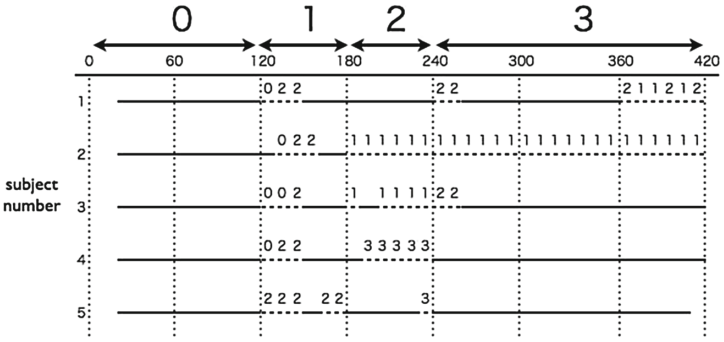


Fig. 7. Erroneous prediction. The numerical figures (0–3) are the erroneously predicted levels. The correctly predicted cases are shown in solid lines.

5 Experiment on Group Sleepiness

Next, we tried to predict the group sleepiness in a simulated classroom. We conducted a similar experiment as the experiment on individual sleepiness. Five subjects were asked to perform a sequence of behaviors, for instance, awake for the first 1 min and then nod for 2 min. They were all different from those attended in the experiment of individual sleepiness. The instruction sequence was designed to have a large variety of combinations of sleepiness levels of five subjects. It is divided into two parts: (1) abrupt sleeping in turn and (2) falling into sleep in

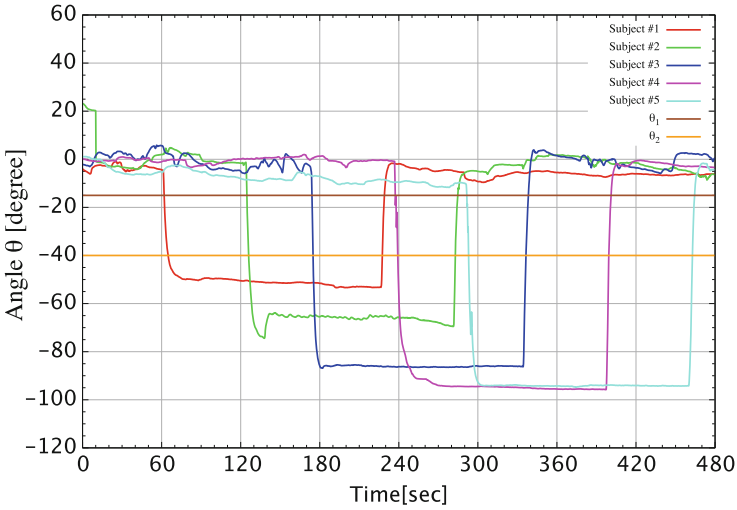


Fig. 8. Observed individual head angles in 0 to 480 s.

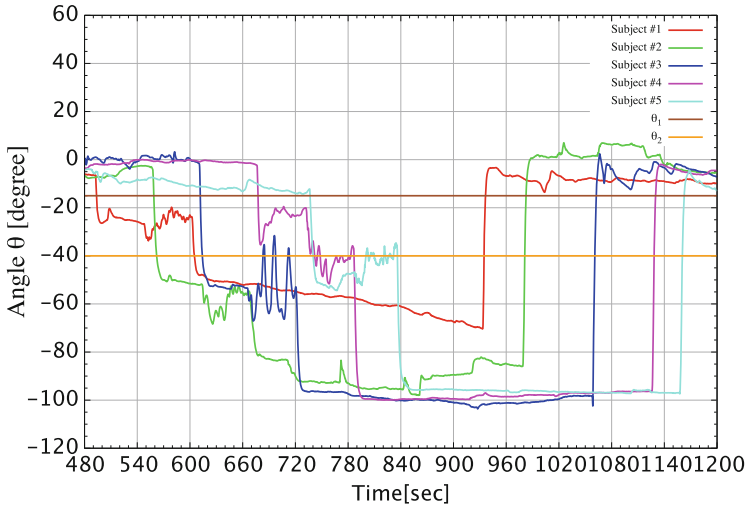


Fig. 9. Observed individual head angles in 480 to 1200 s.

turn. These two parts were carried out sequentially. In the first part, subjects fell into sleep (level 3) suddenly from awake status (level 0) in turn with 1 min interval. Each subject work up from sleep after 3 min, so that at most three subjects slept at the same time. In the second part, every subject, one after the other with one-minute delay, performed all levels from level 0 to level 3 in turn with 1 min interval and all subjects kept sleeping for 5 minutes at last, so that several combinations of sleepiness levels were obtained.

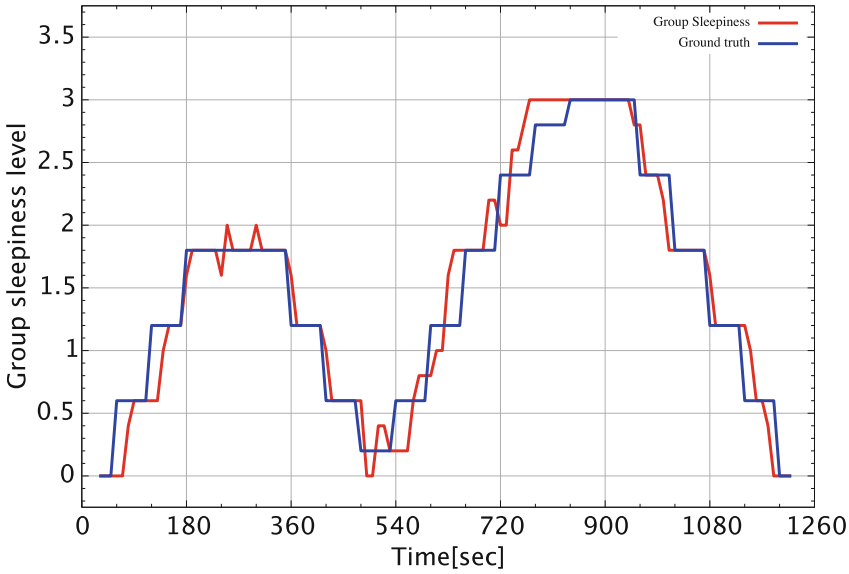


Fig. 10. Group sleepiness. The predicted level (red solid line) and the true level (blue solid line) are compared (Color figure online).

Table 3. Square root of the mean of squared error levels (SRMS)

Subject	SRMS
#1	0.516
#2	0.847
#3	0.742
#4	0.516
#5	0.856
Group (#1–#5)	0.423

The results are shown in Figs. 8–10. Figure 8 shows the result for the first part and Fig. 9 for the second part. In Fig. 8, we can observe that (1) in sleep status (level 3) there is a large variety of head angles depending on subjects, (2) our threshold θ_2 works well for all of them in spite of that the value was determined from different subjects. As a result, as seen in Fig. 10, their sleep status as a group was almost perfectly predicted. In Fig. 9, we can see how they fell into sleep through middle levels. At this time, θ_2 does not seem to work well for some cases. Indeed, in many subjects, the level 2 (drowsy) is classified into level 3 (sleep) for this reason.

The group sleepiness was predicted by the mean of the predicted levels of individual sleepiness. The true level was calculated by averaging the instruction levels. The result is shown in Fig. 10. We can see that (1) overestimation, e.g.,

Fig. 11. Sleepiness visualizer for classroom

level 1 to level 2, happens many times, but the amount is reduced by the effect of averaging compared with individual cases, (2) some errors are seen in the first part (0 to 480 min), but many of them are at the turning points from sleep to awake status of a subject, (3) the prediction of levels larger than the true levels happens in the first half in the second part (480 to 1200 min). In total, the square root of the mean of squared error of levels was 0.431 in the group sleepiness. This is far less than the average 0.695 of individual cases ($0.516 - 0.856$) (Table 3). If we want to know if "awake" or "not", then the prediction is almost perfect.

6 Discussion

Through two simple simulations, we confirmed that measurement of the group sleepiness is easier to that of individual sleepiness, because the averaging reduces the variance. Indeed, in our problem setting, it is more important to know the averaged sleepiness of members in a group than to know who is sleeping or feeling drowsy. Although the prediction rule is very simple using only the mean and variance of head angles during the recent 30 s (150 samples) with some thresholds, the prediction performance of group sleepiness level is rather satisfactory.

The goal of this study is to detect the situations under which many people tend to feel drowsy. The background idea is that there must be some environmental cause if many of members feel drowsy at the same time. Therefore, if we can know when and to what degree people sharing a same situation feel drowsy, it would be useful to avoid possible risks and to reduce inefficiency of labor in somewhere such as factories and classrooms. If a company knows that many workers in its factory feel drowsy at the same time for some time period, they might introduce a refresh time, change the air, strengthen the light, or play music. In a classroom, a teacher could improve the way of teaching if he/she can know when and how many people are about to be loosing their concentration to the lecture. Indeed, we have been developing a visualizer for classroom activity (Fig. 11).

7 Conclusion

We have invented a device to measure individual sleepiness and proposed a way to measure the group sleepiness using those devices connected to each other by wireless. It is better than video cameras, because there is no occlusion by someone else or desks. The maximum number of units is 65,536 and the reachable area is of radius of 30 m, so that the system can be used in a large space such as classrooms and middle-size factories. Some simple simulations revealed that the level of group sleepiness can be predicted more reliably than that of individual sleepiness.

At present, a unit is tied to a bandanna and attached to one side of ears. The size is not so little so that people might feel annoyed by attaching it. It is desired to reduce the size for practical usage. The prediction rule could be improved to raise the performance. The current precision is sufficient to know if many people sleep or not, but not always sufficient to know their drowsy levels.

References

1. Papadelis, C., Kourtidou-Papadeli, C., Bamidis, P.D., Chouvarda, I., Koufogiannis, D., Bekiaris, E., Maglaveras, N.: Indicators of Sleepiness in an ambulatory EEG study of night driving. Eng. Med. Biol. Soc. **1**, 6201–6204 (2006)
2. Chieh, T.C., Mustafa, M.M., Hussain, A., Hendi, S.F., Majlis, B.Y.: Development of vehicle driver drowsiness detection system using electrooculogram (EOG). In: Signal Processing with Special Track on Biomedical Engineering, pp. 165–168 (2005)
3. Norihisa, F., Yoshinori, T., Noboru, O.: Detection of sleeping students using video images. Forum Inf. Technol. **4**, 307–308 (2002). (In Japanese)
4. http://www.kionix.com
5. http://www.digi.com

A Semantic Reasoner Using Attributed Graphs Based on Intelligent Fusion of Security Multi-sources Information

Vincenzo Carletti(✉), Rosario Di Lascio, Pasquale Foggia, and Mario Vento

DIEM, Department of Information Engineering,
Electrical Engineering and Applied Mathematics,
University of Salerno, Fisciano, Italy
{vcarletti,rdilascio,pfoggia,mvento}@unisa.it

Abstract. Recently, the need of monitoring both real and virtual environments is growing up, especially in security contexts. Virtual environments are rich of data produced by human interactions that can not be extracted using classical physical sensors. Thus, new kind of sensors allow to obtain and collect a huge quantity of data from these virtual environment. In order to monitor complex environments, in which the human factor is essential, arises the need of combining both data derived from objective measurements (hard data) and data derived from human interaction (soft data). In this paper we present a method and a software architecture for the fusion of heterogeneous data. The novelty of this method is the joint use of a rule-based inference engine, of a graph matcher and of semantic ontology reasoning to combine and process structured data coming for hard and soft sources. An application of the proposed system is presented within the framework of a Security Intelligence project.

1 Introduction

The term *sensors* is, generally, referred to devices that measure a physical quantity in a real environment and provide a physical signal reflecting the measured value. Then the output signal is interpreted by other instruments. In many real applications the monitored environment is covered by a rich network of sensors, wherein each sensor is employed to grab the state of a specific physical quantity. In order to associate, correlate and combine data coming from each sensor of the network many systems adopt multisensor data fusion technologies, as extensively described in a recent survey of Khaleghi et al. [1].

Nowadays, the need of monitoring a new kind of non real environments is growing up. These new environments are represented by virtual places (such as blogs, forums, online journals, social networks and so on) wherein there are not physical quantities to measure, but interactions among humans. The interest in these environments is related to the need of analyze the human factor, especially for homeland and public security, as discussed in [2,3]. Thus, a new kind of virtual sensors have been created to grab data form virtual places. These sensors

P.L. Mazzeo et al. (Eds.): AMMDS 2014, pp. 73–86, 2014.
DOI: 10.1007/978-3-319-13323-2_7

have been named *soft sensors* to distinguish them form physical sensors, namely *hard sensors*. As described in [3–5], the term *soft* refers to the fact that they work on soft data, i.e. data obtained by processing human observation written in natural language, whose uncertainty is not easy to evaluate. Moreover, different approaches have been proposed also to extract and combine data from multimedia source, for instance in [6,7] the authors propose a method based on a combination of classifiers.

Extracting and combining data obtained by soft and hard sensors represents one of the most challenging topic in the field of Information Fusion and it is a problem still open [1]. Recently, several papers have emphasized the growing interest about this paradigm [4,8–11]. One of the most interesting proposals on this topic is the paper by Gross et al. [5], presenting a new architecture for processing hard and soft data. This architecture uses a fusion framework based on attributed relational graphs (ARGs) and graph matching techniques to handle the representation and the combination of heterogeneous data in an efficient manner. One of the limitations of this method is that it requires some ad hoc preprocessing of the data to solve alignment problems between the data sources (e.g. spatial or temporal alignment); another problem is that the method does not make provisions for the use of context-dependent, a priori knowledge about the application domain in the fusion process.

In this work we propose an innovative system to face the fusion of high level and structured data extracted both from soft and hard sources. Our proposal, as in [5], is based on graph matching; however, we have integrated graph-based algorithms with other techniques in order to overcome its limitations: first, we adopted a rule-based inference engine in order to preprocess the input data by aligning and extending the graph attributes; the rules can be context-dependent, so providing a first way for incorporating background knowledge. Furthermore, we use ontology reasoning for extending the relationships between the input data, with the possibility of context-dependent ontologies. Lastly, we have developed an efficient template matcher for detecting and reporting situations of interest within the combined data. This matcher can be used both for notifying other software components (for instance, to request the attention of a human operator for a potentially dangerous situation) and for injecting new, higher level information in the fusion system, in order to progressively raise the abstraction level of the obtained information.

The proposed system has been developed within the Security and INTelligence SYStem (SINTESYS) Project [12], funded by the Italian ministry of Research and Education, aimed at the realization of an innovative architecture to support investigators in complex security-related scenarios, involving the joint use of several advanced and heterogeneous technologies such as video and audio analytics, text analysis, social network analysis in order to prevent security threats ranging from hooligan riots to terrorism or organized crime.

2 System Overview

The core of the system is the Fusion Center that works on high level structured data, namely *Events*. Each of them describes a simple situation detected by an *Event Source*, i.e an analytic module composed by a sensor network needed to obtain the state of an observed environment that can be both virtual and real. Each event source analyzes raw data coming from the sensors then generates an event containing a summarized high level description of what it happens into the environment. An example of event could be the presence of a given person in a specific place, detected by face recognition module that analyzes the video stream of a surveillance camera. The system overview, in Fig. 1, shows three main typologies of event sources:

- *Audio and speech analysis sources* processes phone calls, environmental microphones to identify people or to detect impulsive audio events like gun shoots, screams or broken glasses.
- *Text analysis sources* extracts data from web pages, journals, social network posts, etc.
- *Video and image analysis sources* processes image or video, for instance to identify people, organizations, symbols, etc.

The Fusion Center collects the incoming events and tries to find the relationships among them in order to realize a complex event network by correlating and summarizing all the information available on the state of the environment. Then, it analyzes this event network by searching for occurrences of a set of known *Templates* in order to discover potential threats or other situations to be notified, either to a human operator or to other subsystems (e.g. for requesting further data). This mechanism can also be used to automatically generate

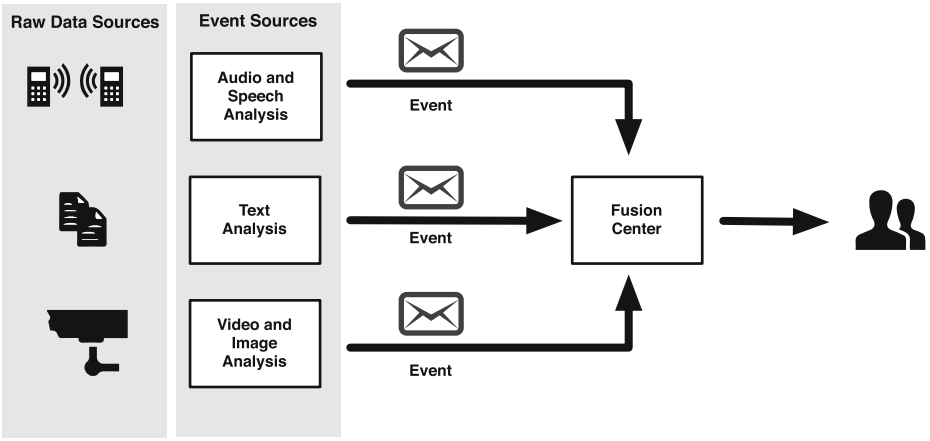


Fig. 1. Overview of the system. The event sources generate events from raw data sources and send them to the Fusion Center. The latter notifies a situation of interest to human operators.

higher-level events to make evident the information that emerges when lower-level events are combined. High-level events are sent back to the Fusion Center, thus they could potentially trigger other discoveries in cascade.

Despite, the process seems to be linear, several issues make the task of the Fusion Center very challenging [1,4,5,13,14]: temporal and spatial alignment, data uncertainty, conflicting and duplicated information. Furthermore, this kind of systems requires to be applied in several environments, then the Fusion Center has to adapt its behavior to the analyzed environment. Finally, several and heterogeneous event sources need to be managed by the Fusion Center: thus, it is not obvious how to represent in a unified and effective way the information coming from these different sources.

2.1 Event Representation

Commonly, hard data is structured using a vector based representation that can make use of a wealth of techniques derived from statistics, machine learning, discriminant analysis and so on. But, its main shortcoming is the unsuitableness to represent the complex structure of information coming from soft data sources, such as the informative content of a text. For this reason, we propose a graph based representation, which allows to handle all the relationships between data. In this representation each node of the graph has a specific type defined into an ontology. This states all the properties and relations a node could have in the graph; main node types are: event node, entity node, place node, context node, attribute node. Moreover, nodes are linked each other by different typologies of relationships, depending on the type of endpoints. One of the main advantages behind the proposed representation is that it can easily contain spatial and temporal information by dedicated nodes. In this way, the system is able to properly deal both temporal and spatial alignment during the fusion process by exactly situating the event both in space and time.

An example is shown in Fig. 2. We consider an event generated by a video analysis source. Round nodes are associated to event (riot1 in Fig. 2), entity (person1 and organization1), place (place1) and context nodes (Violence), while the squares identify attribute nodes. Finally, the role of the entities involved in the event is expressed by the labels of the relationships. In particular, the RDF graph describes a person, Mario Rossi, accused for a riot by the police. The event happens at Kings Cross, in London, in July, 23th at 02:16 pm.

Event Format and Ontology. The events are structured according to the Resource Description Framework (RDF) format. RDF is a W3C standard [15], usually employed in semantic web technologies in order to represent resources distributed on the web as well as the relationships between them. Each object manipulated by the system has to be intended as an RDF resource identified by an Uniform Resource Identifier (URI), except for the attribute nodes. Moreover, the chosen format allows to naturally exploit ontologies. An ontology permits to represent the knowledge by combining concepts within a given domain.

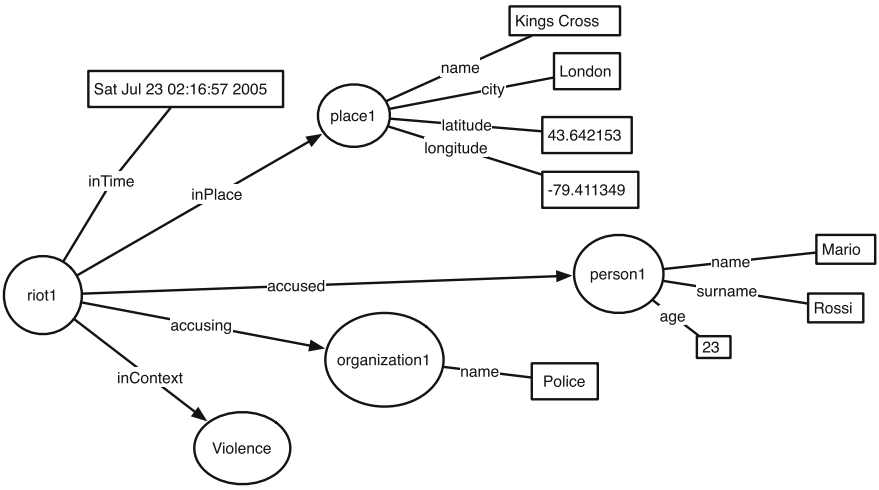


Fig. 2. The RDF graph representation of an event. The labels of round nodes (riot1, place1, person1, organization1) represent the name of instances for the ontology classes the node belongs to. The exception is the Context Node (Violence), whose label is exactly the ontological class. Whereas, the labels of squared nodes represent the value of the attribute; its typology is specified by the label of the edge having as endpoint the Attribute Node.

The knowledge can be easily shared among all the modules composing the system, in terms of typologies of events, entities and relationships. In this way, the system can take advantage form this shared base to perform a more efficient inference on data. In particular, the proposed architecture adopts the Web Ontology Language (OWL) [16], which allows to define an ontology schema for RDF based representation.

The choice of RDF and OWL imposes the use of a triplestore to make the system knowledge base persistent. The triplestore is queried using SPARQL Protocol and RDF Query Language (SPARQL) [17], a query language able to retrieve and manipulate data stored in RDF format.

3 Architecture

The system is based on a distributed architecture: each event source generates and sends events to the Fusion Center. The latter collects and correlates the incoming events in order to enrich its knowledge base and find situations of interest.

Apache Jena [18], a framework for building Semantic Web applications, is the core of Fusion Center. This framework allows to employ all the technologies described in Sect. 2. The communication between the sources and the Fusion Center is guaranteed by a set of RESTFul Web Services, that makes use of an intermediate JSON representation of the events.

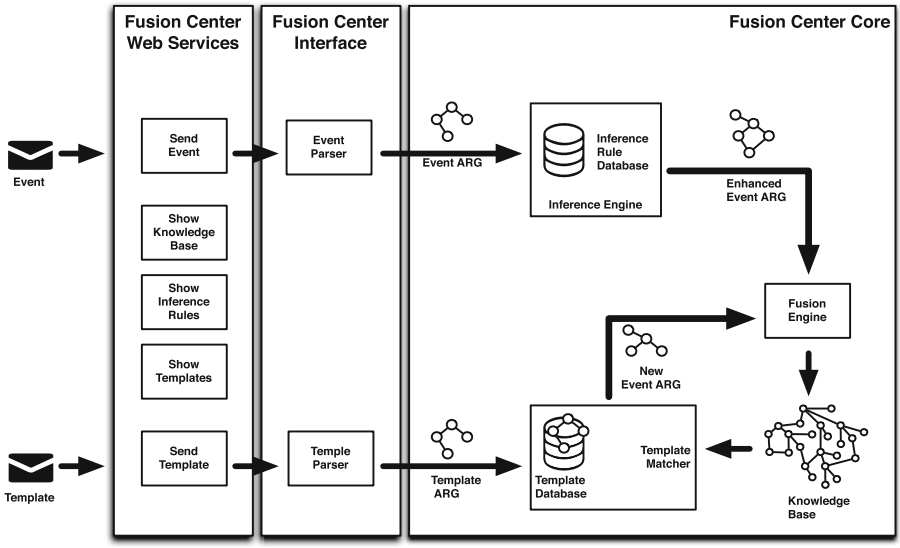


Fig. 3. System modules overview. In this figure the system is divided in three logical sections: Web services, Interface and Core. Web services are used to communicate with the other module of the system, such as the Event Sources. Interface modules are used to translate the format of input/output events. Finally, the Core contains all the modules need for the fusion process.

As regard the structure, the Fusion Center is composed of the following modules (Fig. 3): Event Parser, Inference Engine, Fusion Engine, Template Matcher. Each of them will be descried in details below.

3.1 Event Parser

The event parser constitutes an input/output interface from the Fusion Center to the other modules of the system, and vice versa. As soon as an event comes from a source, the Event Parser reads the content of the JSON string and builds an RDF representation of the event. In the same way, when a new event is generated form the Fusion Center, the parser converts the event from the RDF representation to the JSON one.

3.2 Inference Engine

The Inference Engine aims to extract the implicit information contained inside an event and then adds new relationships among its data. Indeed, raw events coming from the sources are composed of a base informative content that is not completely useful for the fusion engine, so it has to be enhanced. This task is performed employing a set of context dependent rules that could be easily provided by a human operator. An Inference Rule Database allows to store the

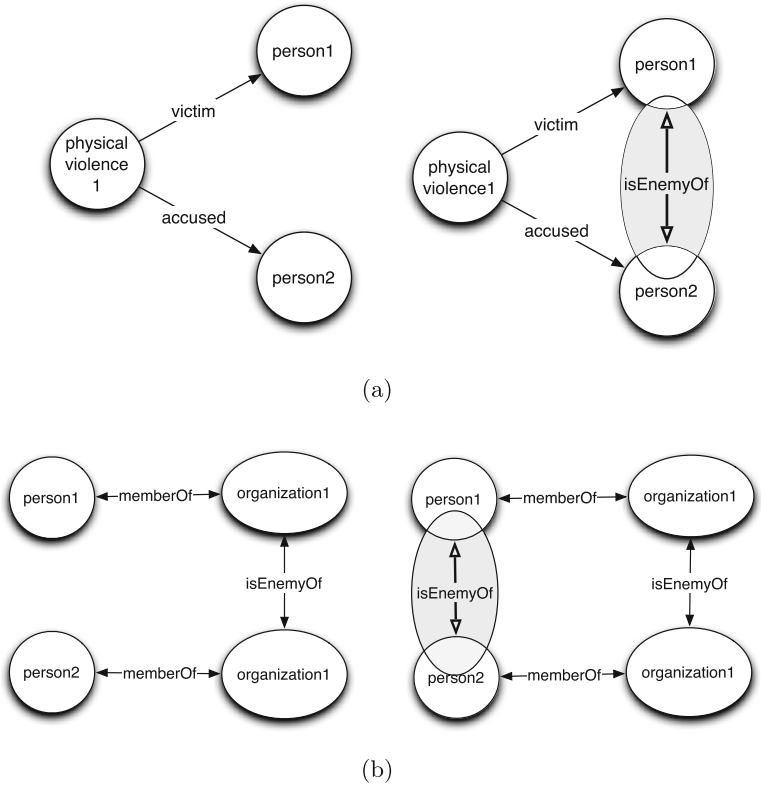


Fig. 4. Examples of relationships extracted by the Inference Engine module. (a) An event states that person1 has used violence on person2; the module infers that they are enemies. (b) The module infers that two members of enemy organizations are also enemies between them.

rules using the Semantic Web Rule Language (SWRL) standard format [19]. As base to realize the inference engine, we adopted Pellet [20], an open source OWL reasoner, due to its efficiency as well as to the fact that it can be easily integrated into the Apache Jena inference subsystem (Fig. 4).

3.3 Fusion Engine

The Fusion Engine is the core module of the Fusion Center. It has in charge of fusing all the incoming event with the ones already contained inside the knowledge base. Events fusion is performed by considering the correlations between two or more events identified by this module. We distinguish between two kinds of event correlation, namely strong and weak, in order to find a criterion to fuse events. A strong correlation is found if two events share the same people or organizations. On the other end, a weak correlation is related to the spatial proximity (stating if two events happen in the same place or in a neighboring areas) or a temporal

closeness. It is evident that, without any strong correlation, a weak correlation is not sufficient to state that two events have to be linked. In order to find strong correlations, we have to define a function to state if two distinct entities correspond to the same entity. The presence of incomplete or incoherent data describing entities makes difficult to precisely identify if two events share the same entity. Thus, we have defined a similarity function that compares each shared attribute of the two entities and provides an entity similarity score. If this score underlies a given threshold the analyzed entities will be considered identical. It means that the Fusion Engine fuses their attributes in a new entity: if the attributes are not shared between the entities, they will be simply added; otherwise the coherence of the attributes value needs to be evaluated and the one corresponding to the lower uncertainty value will be chosen. Furthermore, we use the geodesic distance to compute the distance between two places and we define two thresholds to determine if they are close, far or the same place. Similarly, we evaluate the temporal closeness to ascertain if two events happen at same time or not. The use of thresholds, configurable by a human operator, make us able to handle a tolerance on the spatial and temporal distances between events.

3.4 Template Matcher

After new events have been fused into the system knowledge base, the Template Matcher has to scan it to determine if a situation of interest arises. If so, the Template Matcher produces a new event, which is notified to an human operator and then submitted back to the Fusion Engine. In this way, the new event is processed and may contribute to create a new situation of interest.

Each situation of interest is represented by a model that defines both the pattern to search for and the structure of the new event to build. The pattern is expressed in terms of RDF triples with some unknown elements to find on the knowledge base (Fig. 5). These triples are combined into a SPARQL query, whose result consists in the unknown elements to be reificated and then to be used for completing the information into the new event.

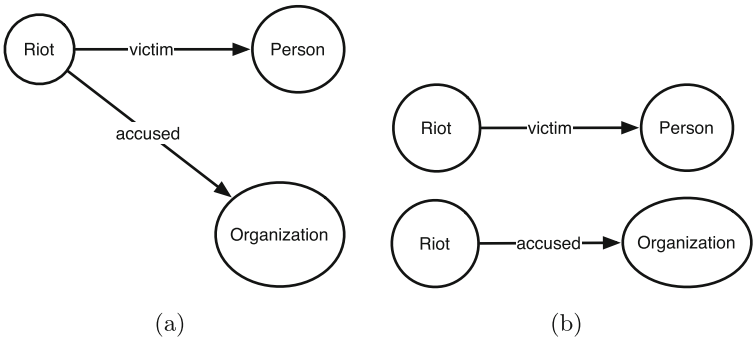


Fig. 5. Example of model pattern. (a) The complete pattern. (b) Pattern triples.

It is important to point out that if a structural pattern is into the knowledge base, it will ever be inside; thus, it will be found by the Template Matcher at each pattern search. Due to this fact, if an event is generated once, it will be sent for ever. A first simple solution to this problem is to store every sent event internally, in order to check if a new-made event has been already generated in the past. The main problem of this approach is the efficiency. Indeed, as the number of stored events increases then even the computational time required to check a new event and the number of the results for the pattern query grows up. To avoid this issue, we have adopted a more efficient solution based on the division of the knowledge base graph in temporal partitions, each of them obtained by an RDF Named Graph. The RDF Named Graph is the RDF triple extended with an extra information representing the temporal epoch it belongs to. In Fig. 6 is shown an example of parted knowledge base.

When the Template Matcher searches for a pattern, it has to check first if at least one of its triples belongs to the latest epoch, than it executes the whole SPARQL query. This preliminary check is performed by querying the current

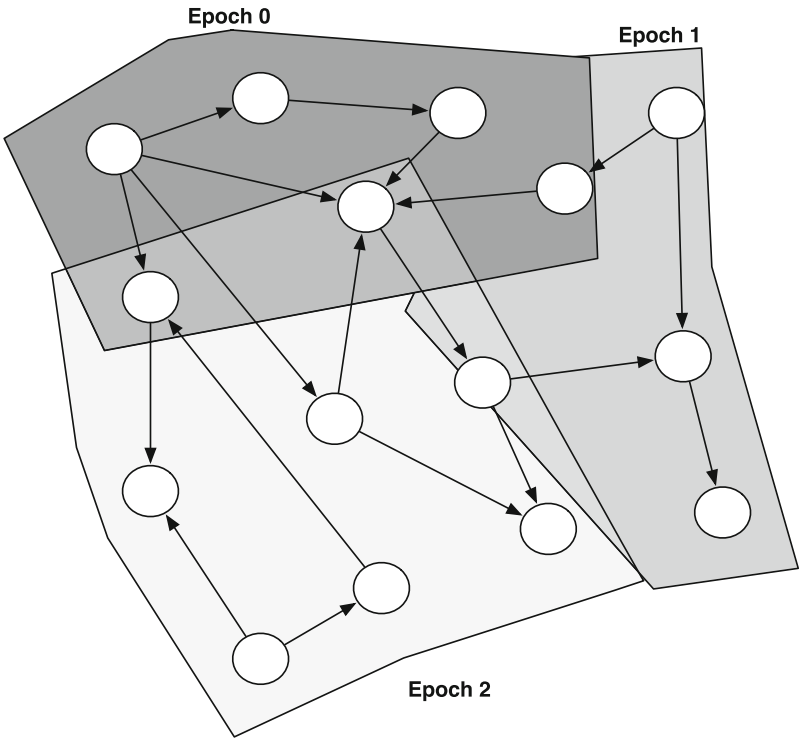


Fig. 6. Example of the knowledge base graph parted in epoch subgraphs. Note that the partitions are built over the triples and not over the nodes. Indeed a node (a subject on and object) can be shared between two or more partitions but the edges have to be contained just in a single subgraph.

partition with each pattern triple separately. In this way, the Template Matcher avoids to search the whole pattern query over the entire knowledge base, even if no new events have to be generated. Finally, if one or more events have been produced, the epoch is increased.

4 Case of Study

In this section we will describe a typical scenario in which the system operates. The scenario represents a typical situation that may happen in a sports context: the enmity between the supporters of two soccer teams degenerating in a riot. Moreover, the effects of the riot spread until a risk for another soccer match, involving other teams, is raised. The Fusion Center collects the events coming, at different time instants, from the event sources; then, it infers new relationships and notifies a risk for an incoming match. Figure 8 shows two different views of the relationships inferred by the Fusion Center both for entities and for events. Due to the complexity of a complete RDF representation of the knowledge base, we prefer to show two separated graphs, so to highlight the main the principal relationships extracted by the Fusion Center.

In the proposed scenario we suppose to search for the pattern in Fig. 7; which describes the structure to find into the knowledge base. If the structure is found, the Fusion Center will notify a possible risk of riot during the match that involves the considered soccer team. Note that the pattern also specifies the context (Soccer in the proposed eaxmple); it is clear that without this information the structure could be generalized for several situations not related to the soccer.

In Table 1 a subset of the events composing the whole scenario is summarized. Due to the impossibility of inserting in this paper all the graphs of the events and for the sake of readability, we decided to show just a few important examples, aiming to understand the fusion process.

In Fig. 8(a) the principal entities involved into the scenario and the relationships between them are shown; each kind of edge represents a typology of

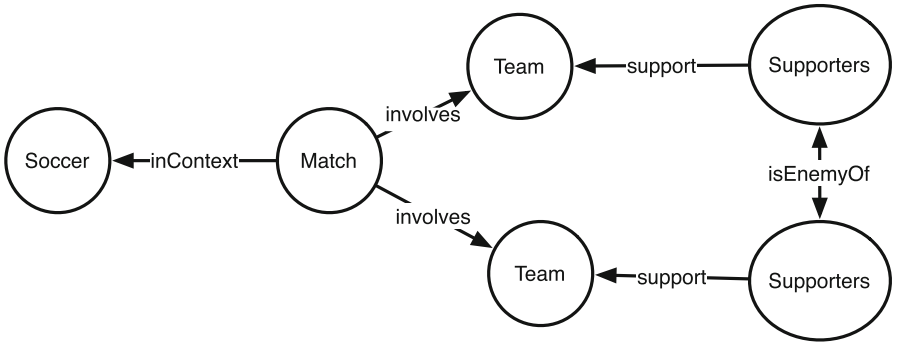
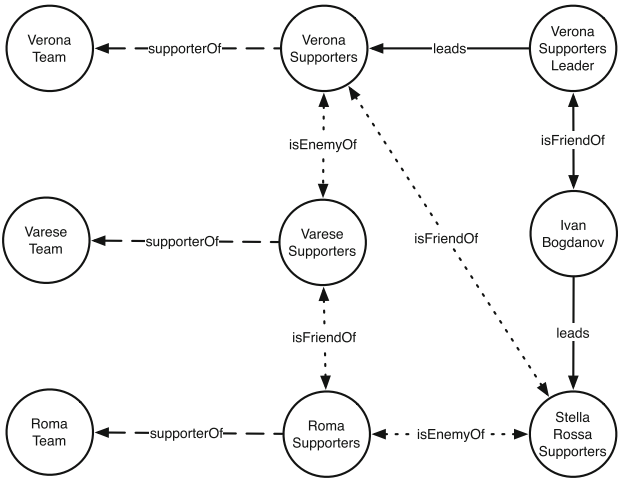
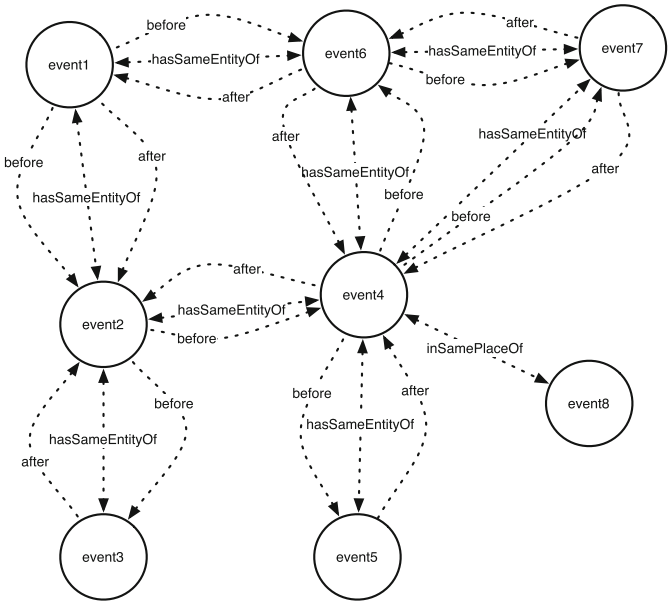


Fig. 7. Graph based representation of the pattern to be searched into the knowledge base



(a)



(b)

Fig. 8. Summarized representation of objects and relationships managed by the Fusion Center. (a) Graph of the entities involved into the events. (b) Graph of the events and their inferred relationships.

Table 1. Events in the considered scenario

Source	Class/Context	Involved entities	Description
Video	Riot/Sport Violence	Verona supporters and Varese supporters	Verona supporters have ambushed a group of Varese supporters. Place: Verona (45.816667, 8.833333)
Image	Relationship/Soccer	Verona supporters and Ivan Bogdanov	In the web site of the Verona supporters is exposed an picture of Ivan Bodganov
Speech	Friendship/Soccer	Leader of Verona supporters and Ivan Bogdanov	The leader of Verona supporters sympathize for Ivan Bogdanov
Text	Leadership/Soccer	Ivan Bogdanov and Stella Rossa supporters	Ivan Bogdanov leads the supporters of the Stella Rossa team
Video	Riot/Sport Violence	Roma supporters and Stella Rossa supporters	Stella Rossa supporters have ambushed a group of Roma supporters during the soccer match. Place: Rome (41.890519, 12.494248)
Text	Relationship/Soccer	Roma supporters and Varese supporters	Roma supporters sympathize for Varese supporters
Text	Threat/Soccer	Roma supporters and Stella Rossa Club	Roma supporters intimidate Stella Rossa Club
Text	Sport Event/Soccer	Roma soccer team and Verona soccer team	A match between Roma and Verona soccer teams is incoming. Place: Rome (41.890519, 12.494248)

relationship: those extracted directly from the event have a solid edge, the inferred relationships have a dotted edge and, finally, those already in the knowledge base have a broken line edge. Note that the enmity and friendship relationships inferred by the Fusion Center are fundamental for the pattern we are searching for.

Furthermore, in Fig. 8(b) the relationships among the events are shown. As soon as a new event comes into the Fusion Center, it tries to fuse the event with the others in the knowledge base. As mentioned in Subsect. 3.3, the Fusion Center searches for a strong correlation first (hasSameEntityOf in Fig. 8(b)), then it tries to add weak correlations, such as temporal correlation (before and after) or spatial correlation (inSamePlaceOf). The event relationships can be used to represent situations more complex than the one we have proposed in the pattern in Fig. 5. In fact, complex situations, more typical in real application, need also to involve spatial and temporal relationships among the events. Think, as an example, to those situations which require a strict temporal causation for happening.

It is important to point out that all the RDF triple, generated during the evolution of the scenario, belong to the same epoch in the knowledge base. Indeed, the pattern in Fig. 7 is found when the last event arrives. Before a new event is generated by the Fusion Center the epoch will not change.

5 Conclusion

In this paper we have proposed an innovative architecture for the fusion of both soft and hard data sources. The architecture is based on the combination of graph-based algorithms with a rule-based inference engine and with semantic technologies such as RFD, OWL and SPARQL. The proposed architecture has been developed within the context of a larger project, SINTESYS, on security-related intelligence from advanced heterogeneous data sources, and has been validated using several application scenarios of this project.

Acknowledgments. This project has been partially supported by MIUR (Italian Ministry of Education and Research) with SINTESYS Project (PON01_01687)

References

1. Khaleghi, B., Khamis, A., Karray, F.O., Razavi, S.N.: Multisensor data fusion: A review of the state-of-the-art. Inf. Fusion, 14(1), 28–44 (2013). http://www.sciencedirect.com/science/article/pii/S1566253511000558
2. Pravia, M., Babko-Malaya, O., Schneider, M., White, J., Chong, C.-Y., Willsky, A.: Lessons learned in the creation of a data set for hard/soft information fusion. In: 12th International Conference on Information Fusion, FUSION '09, pp. 2114–2121, July 2009
3. Pravia, M., Prasanth, R.K., Arambel, P., Sidner, C., Chong, C.-Y.: Generation of a fundamental data set for hard/soft information fusion. In: 2008 11th International Conference on Information Fusion, pp. 1–8, June 2008
4. Hall, D., McNeese, M., Llinas, J., Mullen, T.: A framework for dynamic hard/soft fusion. In: 2008 11th International Conference on Information Fusion, pp. 1–8, June 2008
5. Gross, G., Nagi, R., Sambhoos, K., Schlegel, D., Shapiro, S., Tauer, G.: Towards hard+soft data fusion: Processing architecture and implementation for the joint fusion and analysis of hard and soft intelligence data. In: 2012 15th International Conference on Information Fusion (FUSION), pp. 955–962, July 2012
6. Cordella, L.P., Foggia, P., Sansone, C., Tortorella, F., Vento, M.: A cascaded multiple expert system for verification. In: Kittler, J., Roli, F. (eds.) MCS 2000. LNCS, vol. 1857, pp. 330–339. Springer, Heidelberg (2000)
7. De Santo, M., Percannella, G., Sansone, C., Vento, M.: Unsupervised news video segmentation by combined audio-video analysis. In: Gunsel, B., Jain, A.K., Tekalp, A.M., Sankur, B. (eds.) MRCS 2006. LNCS, vol. 4105, pp. 273–281. Springer, Heidelberg (2006)
8. Digioia, G., Panzieri, S.: Infusion: a system for situation and threat assessment in current and foreseen scenarios. In: 2012 IEEE International Multi-Disciplinary Conference on in Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), pp. 316–323, March 2012
9. Sambhoos, K., Nagi, R., Sudit, M., Stotz, A.: Enhancements to high level data fusion using graph matching and state space search. Information Fusion **11**(4), 351–364 (2010). http://www.sciencedirect.com/science/article/pii/S1566253509000955
10. High-level fusion for intelligence applications using recombinant cognition synthesis. Information Fusion 13(1), 79–98 (2012). http://www.sciencedirect.com/science/article/pii/S1566253510000758

11. Zhang, T., Du, Y.: An information prediction method integrating soft data with hard data. In: 2010 2nd International Conference on Mechanical and Electronics Engineering (ICMEE), vol. 1, Aug 2010, pp. V1-1–V1-5
12. Italian ministry of Research and Education. Sintesys project web page (2013). http://sintesys.eng.it/
13. McMaster, D., Nagi, R., Sambhoos, K.: Temporal alignment in soft information processing. In: 2011 Proceedings of the 14th International Conference on Information Fusion (FUSION), pp. 1–8, July 2011
14. Premaratne, K., Murthi, M., Zhang, J., Scheutz, M., Bauer, P.: A dempster-shafer theoretic conditional approach to evidence updating for fusion of hard and soft data. In: 12th International Conference on Information Fusion, FUSION '09, pp. 2122–2129, July 2009
15. W3C. Rdf standard web page (2013). http://www.w3.org/RDF/
16. W3C. Owl standard web page (2013). http://www.w3.org/OWL/
17. W3C. Sparql standard web page (2013). http://www.w3.org/TR/rdf-sparql-query/
18. The Apache Software Foundation. Apache jena web page (2013). http://jena.apache.org/
19. W3C. Swrl web page (2013). http://www.w3.org/Submission/SWRL/
20. Clark and Parsia. Pellet inference engine web page (2013). http://clarkparsia.com/

Visual Tracking via Sparse Representation and Online Dictionary Learning

Xu Cheng, Nijun Li, Tongchi Zhou, Lin Zhou, and Zhenyang Wu(✉)

School of Information Science and Engineering, Southeast University, Nanjing 210096, China
{xcheng, lnjleo, tchzhou, Linzhou, zhenyang}@seu.edu.cn

Abstract. Sparse representation has been shown competitive performance on single object tracking. In this paper, we extend this technique to tracking multiple interactive objects and present a novel sparse tracker under the tracking-by-detection framework, with saliency detector for objects detection and sparse representation for objects association. Furthermore, we propose an online dictionary learning scheme to capture appearance variations of objects. To avoid using trivial templates, the dictionary contains not only objects templates, but also background information, resulting in more robust estimation. The experiments demonstrate that our approach achieves favorable performance over state-of-the-art algorithms.

Keywords: Multiple objects tracking · Sparse representation · Online dictionary learning · Appearance model · L1 minimization

1 Introduction

As one of the fundamental topics in computer vision, visual tracking plays a key role in numerous practical applications such as video surveillance, activity analysis, human computer interaction and intelligent traffic. Recent years have witnessed significant advances in object tracking with the development of efficient schemes and fruitful applications. A general way to construct a tracking system involves two key factors: motion model and appearance model. A great deal of works has demonstrated that adaptive appearance model plays an important role in achieving visual tracking. In [1], an incremental visual tracker (IVT) that exploits the low dimensional subspace representation to account for the variations of object appearance is presented. Although it performs well in some videos, this method is less robust to the serious occlusions. The visual tracking decomposition (VTD) scheme [2] employs multiple motion and observation models to cover a wide range change of the object appearance. Grabner et al. [3] treat tracking as a binary classification problem and separate the object from background. In [4], an online multiple instance learning (MIL) scheme puts the positive samples and negative ones into bags to train classifier for the object tracking.

Motivated by the sparse representation, which has shown the promising performance on face recognition [5], and then the researchers apply this technique to visual tracking domain [6–13]. The advantage of the sparse representation lies in the robustness to occlusion, background clutter and noises. Bao et al. [7] extend the L1

P.L. Mazzeo et al. (Eds.): AMMDS 2014, pp. 87–103, 2014.
DOI: 10.1007/978-3-319-13323-2_8

tracker [6] to improve the tracking performance as well as reduce the computational cost using Accelerated Proximal Gradient algorithm. Local sparse representation scheme [8–11] is exploited to effectively handle partial occlusion or non-rigid distortion. In [12], multi-task tracking can share information among different tasks to further improve the tracking accuracy. Zhang et al. [14] review the recently proposed trackers based on sparse representation and analyze the advantages of using sparse coding in visual tracking. However, these works ignore the potential of sparse representation for multi-target classification.

Compared with single object tracking, multi-target tracking usually faces three challenges: new object initialization, re-recognition of re-entering targets and object identity switches during the occlusions. Tracking-by-detection approaches [15–24] have become increasingly popular in multiple objects tracking fields. These methods first detect the objects using an offline learned object detector, and then assign the detection responses to the tracked trajectories using different data association methods. To obtain more reliable association results, appearance models are often online learned from a feature pool to distinguish multiple objects globally [18]. In addition, different optimization methods such as K-shortest path [19], Hungarian algorithm [20] and linear programming [21] are presented to effectively infer the best matching among detection responses. In [23], multi-target is formulated as minimization of a continuous energy function to find strong local minima of energy. Pirsiavash et al. [24] employ a cost function to estimate the number of objects and their track births and deaths. Lu et al. [15] explore a mixed proposal distribution of the particle filter to track players. Although such methods may produce better results, they are not necessarily able to differentiate similar appearance and interactive objects.

In this paper, we will focus on aforementioned challenges. We will further exploit potential of sparse presentation for multi-target tracking and present a sparse representation based objects matching scheme with the spatial relative locations of objects. In addition, an online dictionary learning method is proposed for updating the object templates so that each learned template can capture a distinctive part of objects. The dictionary, which contains rich information from different viewpoints and scales, consists of objects and background. So it gets rid of the trivial templates in particle representation.

The rest of this paper is organized as follows. In Sect. 2, we summarize the works most related to ours. Section 3 briefly reviews saliency detector. The proposed algorithm is presented in details in Sect. 4. The experiments are shown and analyzed in Sect. 5 and the paper finishes with conclusions in Sect. 6.

2 Related Work

Object tracking has been extensively researched in the past decade. Some key issues in tracking are that how to associate the objects in different frames and how to update appearance model to capture the changes of objects.

In recently years, many tracking methods which focus on tracking single object are presented in literature. They usually search the object appearance that is most similar to its template to distinguish one object from all other objects and online update object

appearance model to continuously track the object, e.g., IVT [1], VTD [2], Boosting [3] and L1 [6, 7]. Recently, Kalal et al. [25] propose a novel method that consists of tracking, learning and detection components. The tracker keeps up with the object. The detector localizes all observed appearance and corrects the tracker if necessary. The learning estimates the detector's errors and updates it. In [31], an efficient tracker with the SIFT feature points correspondence and multiple fragments, which can handle the partial occlusions and large variations of the object motion, is proposed under the particle swarm optimization (PSO) framework. Superior fragments are updated to adapt the appearance changes.

On the other hand, most association based schemes pay more attention to tracking multiple targets under the tracking-by-detection framework [15–25]. They often associate detection responses produced using an offline learned detector into longer tracks, and find the trajectories of all targets by a global optimal solution. The appearance models of these approaches are often online learned or predefined to distinguish all the objects in the scenario or to discriminate one object with all others [16, 22]. In [32], multiple trackers with different feature descriptors are utilized, and each tracker is implemented with a different feature based on a particle filter. To fuse these independent trackers, authors propose two configurations of tracker selection and interaction to achieve robust visual tracking in dynamic environment changes. In [33], authors show that tracking different kinds of interacting objects can be formulated as a network-flow Mixed Integer Program. Particularly, the presence of the objects which were not initially detected based solely on image evidence can be inferred from the detections of the others. Huang et al. [20] propose a hierarchical association framework to link short tracklets into longer tracks. In [18], a tracklet association strategy is proposed to collect training samples for learning online discriminative appearance model. Note that [17] creates graph for tracking objects, but nodes in the graph are either detection responses or tracklets, and the relationship between nodes denotes affinity of tracklets, not considering the background information. Though such methods have made a great process to achieve the global optimized trajectories for all the objects, they are not necessarily able to differentiate objects with similar appearance. In addition, [17] is an offline approach that integrates multiple cues, while our approach is an online learning approach that adapts the appearance changes of objects to improve the tracking performance.

3 Saliency Detector

Saliency detector exploits a top-down visual saliency model, which considers spatial consistency using CRF model with latent variables and incorporates local context information. Saliency detector, which jointly learns a conditional random field (CRF) and a discriminative dictionary, is offline trained in CRF framework, which proves to be effective for objects detection.

Given an image, we need to know whether and where the objects appear. We first sample image patches of 64×64 pixels by shifting 16 pixels. Then the Scale Invariant Feature Transform (SIFT) [26] are extracted from all the patches to represent the object appearance. We assign a binary label y to indicate the presence ($y = 1$) or absence

($y = -1$) of the object. Labels can carry the information of the object presence. A patch is labeled as positive if half of its total pixels are foreground; otherwise it is labeled as background. The latent variables c_i is usually obtained to model sparse representation of x_i by the following expression.

$$\mathbf{C} = \arg\min_{\mathbf{C}} ||\mathbf{X} - \mathbf{DC}||_{\mathrm{F}} + \lambda||\mathbf{C}||_1 \tag{1}$$

where $\mathbf{C} = [c_1, c_2, \ldots, c_n]$ denotes the latent variables for all the patches. $\mathbf{D}$ and $\mathbf{X} = [x_1, x_2, \ldots, x_n]$ represent the dictionary and set of patches, respectively. The visual information contained in the dictionary is transferred into the latent variables by $\mathbf{C}$, so it is more informative than image patches $\mathbf{X}$.

In CRF model, a four-connected graph $G = <V, E>$ is built on the sampled patches based on their spatial adjacency. The labels $\mathbf{Y}$ for all the patches enjoy the Markov property on the graph G conditioned on the sparse latent variables $\mathbf{C}$. There, CRF model can be formulated as

$$E(\mathbf{D}, \mathbf{Y}, \mathbf{w}, \mathbf{C}) = \sum_{i \in V} \phi(c_i, y_i, w_1) + \sum_{(i,j) \in E} \psi(y_i, y_j, w_2) \tag{2}$$

where $\mathbf{w} = [w_1, w_2]$ is a weight vector which is trained by the max-margin approach [27]. The energy is measured for each node by $\phi(c_i, y_i, w_1) = -y_i w_1^{\mathrm{T}} c_i$. For each edge, data independent smoothness is considered as $\psi(y_i, y_j, w_2) = w_2 \mathbf{1}(y_i, y_j)$, and $\mathbf{1}$ is an indicator function equaling one for different labels. Finally, similar with the cutting plane algorithm [34], the labels $\mathbf{Y}$ for all the patches in an image can be solved by

$$\mathbf{Y} = \arg\min_{\mathbf{Y}} E(\mathbf{Y}, \mathbf{C}, \mathbf{w}) \tag{3}$$

From Eq. (3), we can know whether and where the objects appear via the labels of patches which indicate the presence ($\mathbf{Y} = 1$) or absence ($\mathbf{Y} = -1$) of the object. Figure 1 shows the detector output. Due to constraints of space, more details can be found in [28].

Fig. 1. The output of saliency detector

4 Proposed Tracking Algorithm

In this paper, we take the popular tracking-by-detection framework. Objects from each frame are first detected by saliency detector and then matched between the consecutive frames. Finally, the dictionary update scheme significantly improves the tracking performance. The proposed algorithm is schematically shown in Fig. 2.

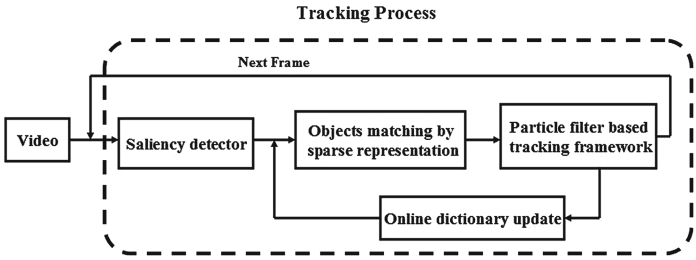


Fig. 2. Overview of the proposed tracking algorithm

During the tracking, the particles of each object in the scene are randomly sampled based on the previous tracking result of the object, and each particle is regarded as a candidate object state. Then we score each candidate object state by the total number of patches which fall into it and belong to the object, and treat the highest-scored state as the tracking result. Therefore, those superior patches (within the tracking result), which indicate the higher appearance similarity with the templates and low possibility of occlusions, are used to continuously track the object.

4.1 Appearance Model for Each Object

The matrix $\mathbf{D} \in \mathbb{R}^{d \times n}$ is the dictionary trained from the detector, and we associate atoms (each column) in the dictionary with the objects in the first ten frames of a video to know which object the atoms correspond to. In other words, the dictionary of each object needs to be constructed to model its appearance. One object is represented by the linear combination of the most similar atoms in the dictionary; then these atoms are assigned the corresponding object index. The atoms which are not assigned any object index are regarded as background. During the process, N sub-dictionaries ($\mathbf{D}_i, i = 1, 2, \cdots, N$) corresponding to $(N - 1)$ labels of objects and one label for the background are obtained from the trained dictionary $\mathbf{D}$, and each sub-dictionary has a corresponding object index. We see that one atom in the dictionary may be used to represent the different objects due to the similar appearance or cluttered background. Therefore, the same atom may also be included in different sub-dictionaries. In other words, each atom in the dictionary is assigned to at least one sub-dictionary which is used to represent the corresponding object sparsely.

In this article, we obtained sparse coefficients of each object by Group-OMP scheme [29] which extends the OMP procedure to handle group selection. The procedure picks the best group in each iteration, and it then re-estimates the sparse

coefficients. The group based greedy pursuit method, which seems more robust for multiple objects sparse classification, is proposed as follow.

$$\alpha = \arg\min_{\alpha} \left\{ \left\| x - \sum_{i=1}^{N} \mathbf{D}_i \alpha_i \right\|_2 + \hat{\lambda} \sum_{i=1}^{N} \|\alpha_i\|_2 \right\} \tag{4}$$

where $\hat{\lambda}$ is a penalty parameter; x and α are one object state and the corresponding sparse coefficient, respectively. The columns of $\mathbf{D}$ are operated as group in each greedy pursuit step. Generally, group based algorithms seem more robust for test samples with burst error.

4.2 Sparsity Based Multi-target Matching

In this subsection, we will introduce an effective object matching scheme between two adjacent frames using sparse representation.

Given an object state x at current frame, the association between consecutive frames can be achieved via Eq. (5).

$$\min_{\mathbf{D}_i} ||x - \mathbf{D}_i \alpha||_2 \quad 1 \le i \le N \tag{5}$$

The index of sub-matrix that corresponds to the minimum reconstruction tracking error is regarded as the object's index. Finally, all the objects can be classified into the corresponding object class by this means.

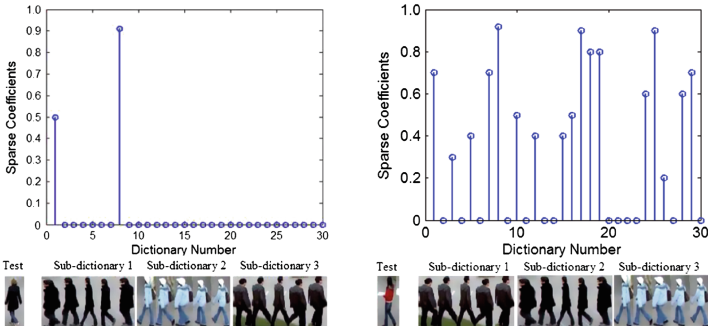


Fig. 3. The association for one test object. (a) One object (left) is linearly represented by one of the sub-dictionary. The maximum of sparse coefficient corresponds to the most similar object. Then, the index of object for two frames is obtained based on the represented sub-dictionary; (b) The nonzero coefficients for a new object (right) tend to scatter among the dictionary.

Figure 3(a) shows the process of the association for one test object. There are three sub-dictionaries, and each sub-dictionary consists of 10 atoms. We try to search the most similar atoms in one sub-dictionary $\mathbf{D}_i$ with a minimum reconstruction error for

the test sample. We can observe that all elements of sparse coefficient are zero except those corresponding to the most similar atoms in the sub-dictionary. In other words, it means that maximum entry in sparse coefficient α corresponds to the most similar object. So we use the index of sub-dictionary which corresponds to the maximum of α to know which object the sample belongs to in the last frame.

In addition, the nonzero entries of α in Fig. 3(b) may be prone to scatter among the dictionary rather than focus on a certain object sub-matrix when a new object appears. In this case, the new object is able to be defined and its corresponding object template is also added to the dictionary **D**.

4.3 Occlusion and Background Clutter

Occlusion and background clutter frequently occur among the objects in the tracking procedure. In this paper, we present a simple but efficient criterion to detect the occlusion in Eq. (6). We set a threshold θ ($\theta = 0.5$ in our article) to detect occlusions.

$$\frac{area(R_i \cap R_j)}{area(R_i \cup R_j)} > \theta,\ i \neq j \tag{6}$$

where R_i and R_j denote the states of two objects, respectively. The occlusion occurs if the overlap rate of two states is above the threshold θ.

The mechanism of saliency detector is based on the local information model, so an object state is composed of patches which describe different sections of the object. SIFT feature points of each patch match with the corresponding template patch. The patches with more matched feature points are used to handle partial occlusion. The relative position of each SIFT keypoint and object center for each object is memorized at frame $t - 1$. At the time t, we can achieve the prediction of the object position by the memorized relative positions of matched feature points. Our tracker can make full use of the partial information (patches) which is not occluded to track the objects. For background clutter, we can also track the object with the patches of more matched SIFT points when there are objects with similar color.

More importantly, we also take the spatial position relationship (2-dimensional coordinates) between the objects into account for keeping up with the indistinguishable objects. It is worth mentioning that the changes of the objects between the consecutive frames are usually gradual due to the mechanical movement (e.g., a person takes off an overcoat during the walking). Therefore, the location information is important to discriminate objects sharing similar appearance or full occlusion. Our approach can take full advantage of the spatial relationship of the objects, but not limited to the partial information of the object, solving the problems of occlusion and cluttered background. The importance of the spatial relationship is reflected in Fig. 8.

4.4 Online Dictionary Update

The appearance of an object in the tracking may change drastically due to intrinsic (e.g., shape and pose variations) and extrinsic factors (e.g., occlusion, illumination changes, and camera motion). A time-invariant appearance template cannot adapt the

appearance changes and may lead to drift. Therefore, it is necessary to update the dictionary in a fixed length frame interval L (L = 10 in our experiments).

In this paper, we propose a stochastic gradient descent scheme to achieve online dictionary updating and consider the acquired dictionary by saliency detector as the initial dictionary. We explore the state with the minimum error in each fixed interval for updating the dictionary. However, the process of updating is paused if the minimum reconstruction error $\varepsilon_{\hat{t}}$ is above the predefined threshold T_0 (T_0 is empirically set to 0.6).

$$\hat{t} = \underset{t\in[(j-1)L+1,jL]}{\arg\min} \quad \varepsilon_t \quad j = 1, 2, \cdots \tag{7}$$

The dictionary $\mathbf{D}$ is not explicitly defined in Eq. (2), but implicitly in Eq. (1). So we exploit the chain rule to compute the gradient of energy function with respect to $\mathbf{D}$.

$$\frac{\partial E_{\hat{t}}}{\partial \mathbf{D}} = \sum_{i\in V} \left(\frac{\partial E_{\hat{t}}}{\partial \mathrm{c}_i}\right)^{\mathrm{T}} \frac{\partial \mathrm{c}_i}{\partial \mathbf{D}} \tag{8}$$

Similar with [30], we establish the fixed point equation of Eq. (1) to overcome implicitly differentiation of sparse code c with respect to the dictionary $\mathbf{D}$.

$$\mathbf{D}^{\mathrm{T}}(\mathbf{D}\mathrm{c} - \mathrm{x}) = -\lambda \mathrm{sign}(\mathrm{c}) \tag{9}$$

where sign(c) presents the sign of c and sign(0) = 0. Then the derivative of $\mathbf{D}$ on both side of Eq. (9) is calculated as follows:

$$\frac{\partial \mathrm{c}_\wedge}{\partial \mathbf{D}} = (\mathbf{D}_\wedge^{\mathrm{T}}\mathbf{D}_\wedge)^{-1}\left(\frac{\partial \mathbf{D}_\wedge^{\mathrm{T}}\mathrm{x}}{\partial \mathbf{D}} - \frac{\partial \mathbf{D}_\wedge^{\mathrm{T}}\mathbf{D}_\wedge}{\partial \mathbf{D}}\mathrm{c}\right) \tag{10}$$

To simplify the gradient computation in Eq. (10), an auxiliary variable s is introduced for each c.

$$\mathrm{s}_{\bar{\wedge}} = 0, \ \mathrm{s}_\wedge = (\mathbf{D}_\wedge^{\mathrm{T}}\mathbf{D}_\wedge)^{-1}\frac{\partial E_{\hat{t}}}{\partial \mathrm{c}_\wedge} \tag{11}$$

where the symbol $\wedge$ denotes the index set of non-zero codes of c and the $\bar{\wedge}$ is the index set of zero codes.

Therefore, the gradient of $E_{\hat{t}}$ with respect to $\mathbf{D}$ is computed using

$$\frac{\partial E_{\hat{t}}}{\partial \mathbf{D}} = -\mathbf{D}\mathbf{S}\mathbf{C}^{\mathrm{T}} + (\mathbf{X} - \mathbf{D}\mathbf{C})\mathbf{S}^{\mathrm{T}} \tag{12}$$

The online dictionary update is performed via Eq. (13).

$$\mathbf{D}_n = \mathbf{D}_l - \gamma\frac{\partial E_{\hat{t}}}{\partial \mathbf{D}} \tag{13}$$

where γ is the learning rate (0.01 in our work). $\mathbf{D}_n$ and $\mathbf{D}_l$ denote the updated dictionary and the last dictionary, respectively.

4.5 Summary of the Proposed Algorithm

The proposed algorithm is summarized as follows.

Algorithm 1. The proposed tracking algorithm

Input: trained dictionary **D**; frame F_t;
Output: objects state $\mathbf{S}_{t,i}$ (the symbols t and i denote the ith object at frame t)

Initialization:
1. Frame t=1: mark all the objects by saliency detector.
2. Run the first ten frames of a video to construct N sub-dictionaries $\{\mathbf{D}_i\}_{i=1,\ldots,N}$ from the dictionary **D**.

Tracking stage:
for t=11 to the end of the sequence
1. Detect all the objects in the scenario by a saliency detector.
for i=1,…,N-1 (N-1 is the number of objects)
2. Randomly generate samples $\mathbf{S}_{t,i}^{p}$ (p denotes the pth particle) from previous result $\mathbf{S}_{t-1,i}$.
3. Score the total number of object patches within each candidate sample and treat the highest-scored sample as the tracking result $\mathbf{S}_{t,i}$.
4. The association of the object is conducted between two frames via Eq.(5).
5. **if** the occlusion or background clutter occurs
6. The object can be continuously tracked by the partial information of object and spatial relationship. (Section 4.3)
7. **end if**
end for
8. Update the dictionary every L frames via Eq.(13).
end for

5 Experimental Results

5.1 Experiment Settings

We evaluate our method on public data sets: CAVIAR and ETH mobile pedestrian. All data used in our experiments is publicly available. For a fair comparison, we use the same parameter setting and adopt the commonly used criteria in [17]: recall & precision are defined as detection performance; mostly tracked (MT) and mostly lost (ML), the ratio of tracked trajectories, which are successfully tracked for more than 80 % or less than 20 % respectively; Fragments (Frag), the number of times that a ground truth trajectory is interrupted; identity switches (IDS), the number of times that a tracked

trajectory changes its matched identity. For these items, recall, precision and MT indicate better results with higher scores; while ML, Frag and IDS indicate better results with lower scores.

In this section, we compare our algorithm against ten state-of-the-art trackers. These trackers are IVT [1], VTD [2], MIL [4], L1 [7], and TLD [25] which are designed for single object tracking, and we extend them to tracking multiple objects in this work. The process can be achieved as follows. Each object is processed by single object tracking algorithm in multiple objects scenarios. In the first frame, we initialize all the targets. The features of each object are extracted and regarded as the appearance template. When a new frame arrives, each object independently runs single object tracking scheme. Then we need to match the features of each object with all object templates and maximum similarity score is a good match to an object. In addition, the change of the object between the consecutive frames is usually gradual due to the mechanical movement. The two constraints of multiple targets tracking scenarios can keep up with objects. When one occurs with occlusion, other targets are regarded as background. The scheme which handles occlusions among the objects is similar to the single object occlusions processing method. Due to space limitations of the paper, more results on the performance of single object trackers are only described in quantitative section. In addition, multi-object tracking methods together with ours are evaluated, such as CEMT (Continuous Energy Minimization Tracking) [23], GOGT (Globally-Optimal Greedy Tracking) [24], BCL1 [11], OLDAMs [18] and CRF Tracking [17]. Some source codes are provided by the authors' websites and the recommended parameters are set for fair comparison.

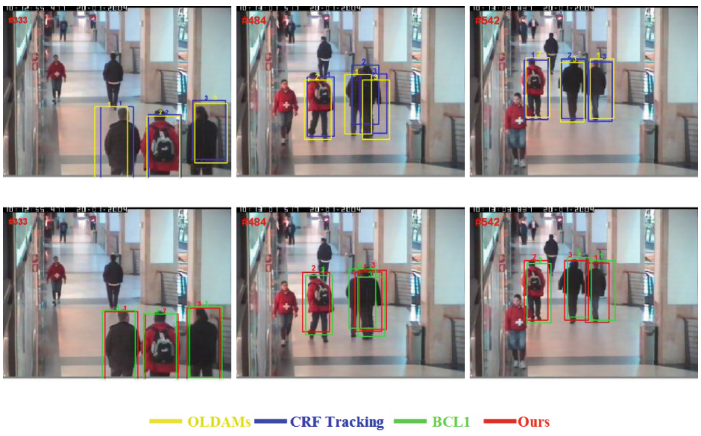
5.2 Qualitative and Quantitative Analysis

The CAVIAR dataset which captures people moving in a shopping center with frequent occlusions and interactions is commonly used dataset for multi-object tracking. We choose three video sequences on this dataset which are the relatively challenging parts of the dataset. The comparison results are shown in Fig. 4 and Table 1. We can see that

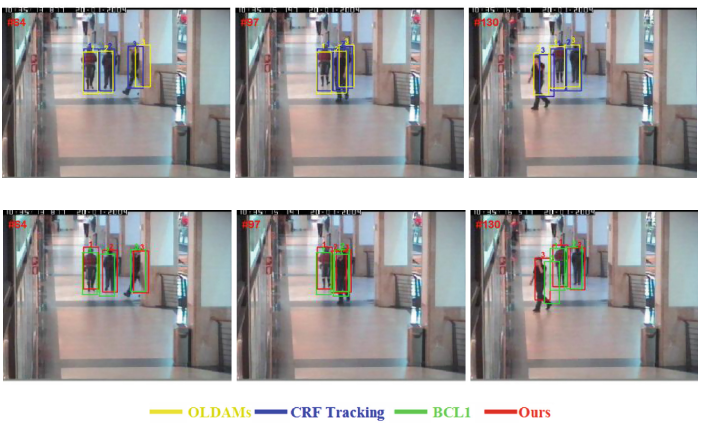
Table 1. Comparison of results on CAVIAR dataset

Algorithms	Recall	Precision	MT	ML	Frag	IDS
ILT	–	–	42.5 %	28.4 %	11	8
VTD	–	–	44.7 %	25.1 %	8	6
MIL	–	–	42.8 %	22.3 %	8	6
L1	–	–	10.2 %	68.0 %	14	11
BCL1	–	–	78.2 %	1.5 %	2	0
TLD	56.0 %	62.3 %	44.8 %	22.9 %	9	6
CEMT	78.9 %	84.2 %	80.1 %	1.5 %	2	0
GOGT	80.1 %	82.0 %	77.4 %	1.0 %	2	1
CRF	78.2 %	82.4 %	79.0 %	0.7 %	3	1
OLDAMs	75.4 %	81.6 %	77.2 %	5.2 %	6	3
Ours	84.2 %	84.7 %	82.4 %	1.2 %	2	0

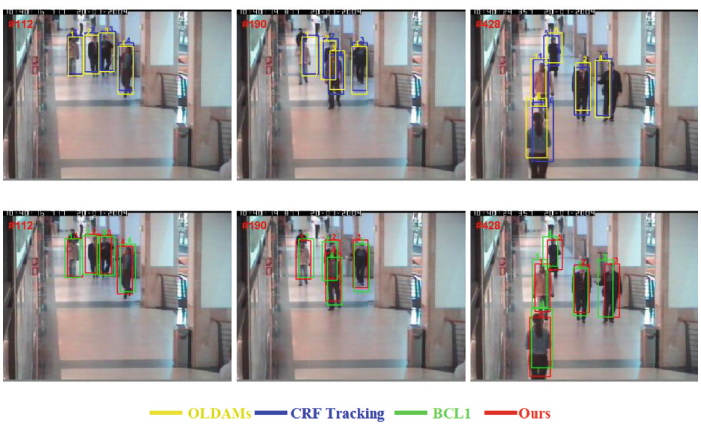
Note "–": The trackers can not meet the criteria definition due to not using the detector.



(a) ThreePastShop2Cor sequence



(b) CAVIAR1 sequence



(c) CAVIAR2 sequence

Fig. 4. Some representative tracking results for CAVIAR dataset.

our scheme does not confuse their identities and finds the correct associations by sparse matching approach. The improvement for our method is a result of two factors. First, the part based idea can better handle the occlusions of objects. Second, the online dictionary update scheme significantly improves the tracking performance. In Fig. 4(b), BCL1 tracker gradually loses the third object. The reason is that covariance feature based can not effectively handle the similar appearance. Other trackers perform well at the beginning of videos. However, they often switch the objects identities when occlusion and interaction occur with similar background clutter and detector sometimes cannot detect them due to some limitation, leading to the unsatisfactory results.

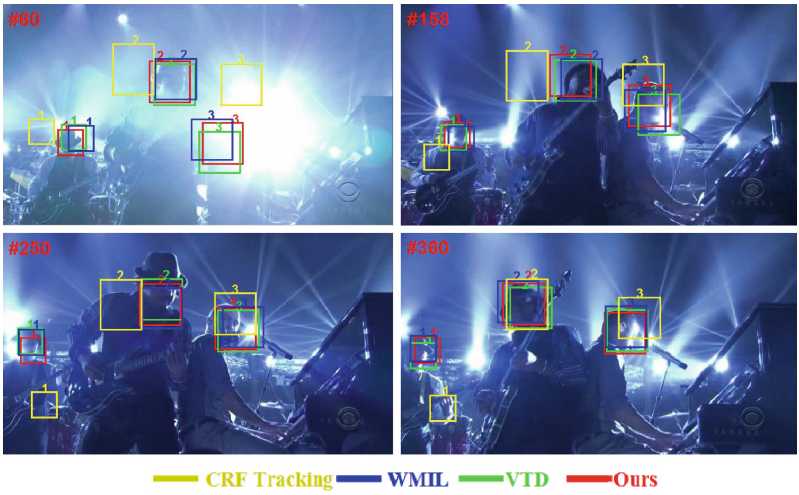


Fig. 5. Tracking results of Shaking video.

The Shaking sequence is captured in a head shaking environment with abrupt object motion and lighting change drastically. The sequence is originally from [2], which is used to evaluate the single object tracking. In our article, we catch up with three objects in the video and compare our approach with other trackers under the circumstance of drastic illumination variations. The single object trackers are extended to tracking multiple objects. Figure 5 presents the qualitative results. We can see that MIL, VTD and Ours can keep up with the objects through the whole sequence. They help a lot in distinguishing the true object from distracters in the background. The remaining trackers start drifting at the beginning of the sequence. The reason is that the appearance update scheme is not easy to capture the changes of objects due to the inaccurate detection.

CAVIAR data is relatively easy; we further evaluate our approach on two sequences of ETH dataset which present frequently heavy occlusions and interaction, illumination changes and many irregular motions. The objects are captured by a pair of cameras on a moving stroller in busy street scenes. The stroller is moving forward, which makes motion more complex.

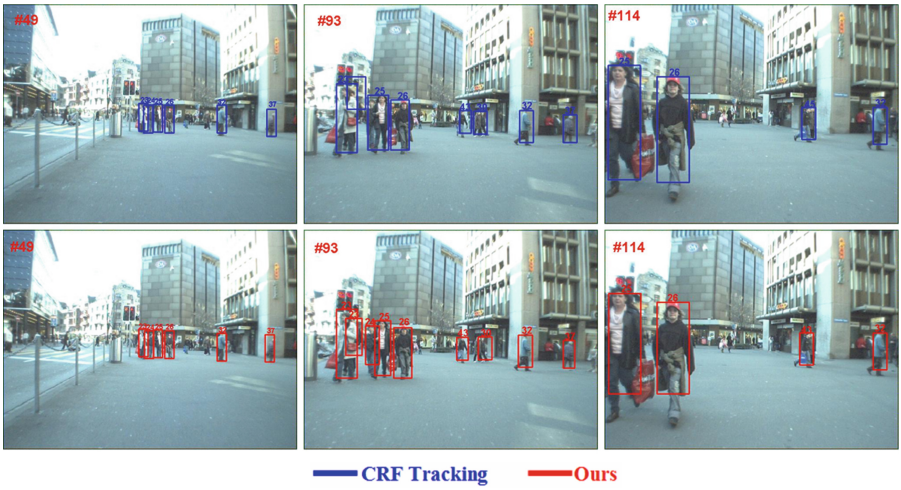


Fig. 6. A representative tracking result for ETH sequence (More results analysis are shown in Table 2 due to the constraint of space).

Some visual results are presented in Fig. 6. It is clear that traditional trackers cannot well connect the objects in neighboring frames. The reason is that these methods based on tracking-by-detection are highly dependent on the detector. However, the results of detection are not often reliable, and even some results are false. Therefore, the association of the objects may fail caused by the inaccurate detection results. To overcome this problem, our approach exploits not only the detection, but also the spatial relative locations of objects to correct the inaccurate detection. With the help of sparse matching scheme, we successfully associate objects into one.

Furthermore, we choose the "sunny day" video sequence for evaluation and use the sequence from the left camera. There are 454 frames, and people undergo frequently occlusions. The first row in Fig. 7 shows that our approach can successfully track object 13 and 14 through the whole sequence, while CRF tracking fails sometimes to find the correct associations due to the bad performance of detector. In Table 2, the obvious improvement in MT, Frag and IDS scores indicates that our tracker can better track objects under the moving cameras and illumination variations, where other trackers are less reliable.

In Fig. 8, the average evaluation scores of our method and ours without the spatial relationship of the objects are presented. Compared with our method not using spatial information, the MT is improved by 10 %, and fragments and identity switches are reduced by 17 % and 11.4 % respectively using the spatial relative locations, while precision and recall scores have slightly improved. This explains the obvious improvements on fragments and identity switches. Therefore, the spatial relative positions are helpful for association the same object in different frames. (Note: FAF denotes the average false alarms per frame.)

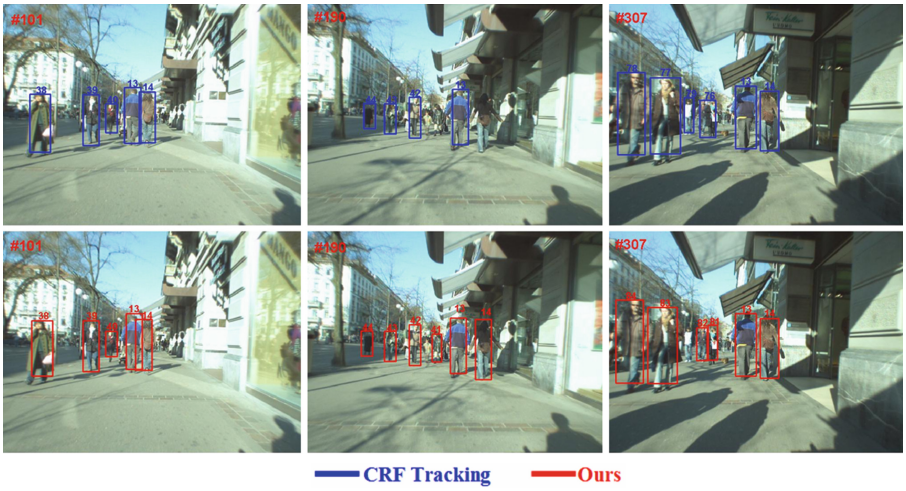


Fig. 7. Long-term tracking results of the couple using our method and CRF tracking under the illumination changes (More results analyses are shown in Table 2 due to the constraint of paper space).

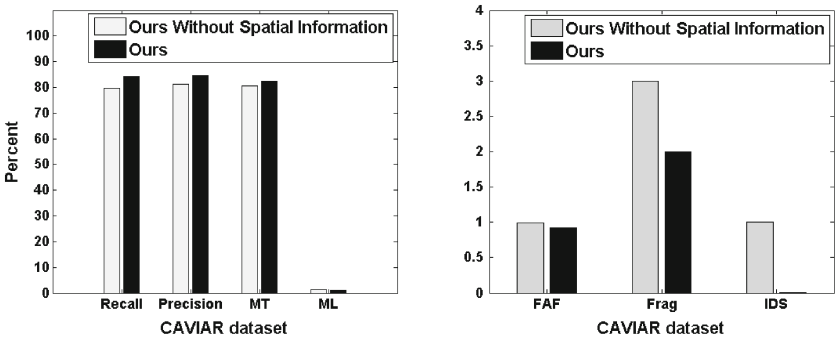
Table 2. Comparison results on ETH dataset.

Algorithms	Recall	Precision	MT	ML	Frag	IDS
ILT	–	–	4.8 %	82.4 %	74	71
VTD	–	–	7.5 %	74.3 %	51	51
MIL	–	–	8.6 %	78.5 %	63	57
L1	–	–	5.4 %	80.0 %	57	57
BCL1	–	–	65.3 %	6.9 %	16	7
TLD	39.3 %	44.7 %	9.1 %	69.7 %	51	48
CEMT	80.1 %	78.4 %	69.4 %	6.8 %	20	12
GOGT	74.8 %	77.1 %	60.5 %	13.7 %	17	15
CRF	79.0 %	88.6 %	68.0 %	7.2 %	14	9
OLDAMs	76.8 %	86.6 %	58.4 %	8.0 %	24	12
Ours	78.3 %	89.4 %	76.8 %	5.1 %	11	5

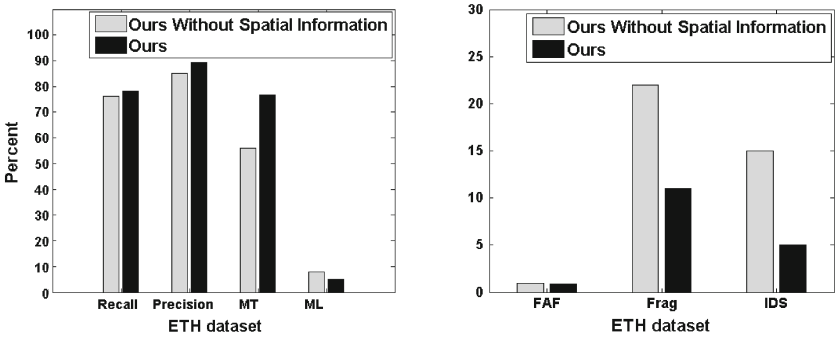
Note "–": The trackers can not meet the criteria definition due to not using the detector.

5.3 Computation Speed

Our experiments are performed on an Intel 3.3 GHz PC with 8 G memory and implemented in MATLAB. The speed relies on the number of objects and particles in videos. The average run time per frame is 3.8 s and 6.2 s for CAVIAR and ETH, respectively. Comparing 3.1 s and 6.0 s for CAVIAR and ETH in [17], most of the computation for our method is spent on SIFT feature extraction and dictionary learning. However, detection time is not included in the measurements.



(a) CAVIAR dataset



(b) ETH dataset

Fig. 8. Comparison of our scheme without spatial information and ours.

6 Conclusion

In this paper, we exploit the sparse representation for multi-object tracking under the tracking-by-detection framework. First, objects are detected in the scenes using the saliency detector. Then sparse representation scheme based can achieve the association with the objects between two frames. An online dictionary updating method is explored to capture the appearance changes of objects. The experiments show that our method improves the performance compared with the state-of-the-art algorithms.

Acknowledgment. We sincerely thank the Computer Vision Lab, University of Southern California (http://iris.usc.edu/people/yangbo/downloads.html) for providing data and ground truth labels.

The authors would like to thank the anonymous reviewers for useful and constructive comments that help improve the quality of this paper. This work is supported by National Nature Science Foundation of China (NSFC) under Grant (No. 60971098, 61201345 and 61302152), the Beijing Key Laboratory of Advanced Information Science and Network Technology (No. XDXX1308) and Priority Academic Program Development (PAPD) of Jiangsu Higher Education Institutions.

References

1. Ross, D., Lim, J., Lin, R.S., Yang, M.H.: Incremental learning for robust visual tracking. IJCV **77**(1), 125–141 (2008)
2. Kwon, J., Lee, K.M.: Visual tracking decomposition. In: CVPR, pp. 1269–1276 (2010)
3. Grabner, H., Grabner, M., Bischof, H.: Real-time tracking via on-line boosting. In: BMVC, pp. 47–56 (2006)
4. Babenko, B., Yang, M., Belongie, S.: Visual tracking with online multiple instance learning. IEEE Trans. PAMI **33**(8), 1619–1632 (2011)
5. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. IEEE Trans. PAMI **31**(2), 210–227 (2009)
6. Mei, X., Ling, H.: Robust visual tracking and vehicle classification via sparse representation. IEEE Trans. PAMI **33**(11), 2259–2272 (2011)
7. Bao, C., Wu, Y., Ling, H., Ji, H.: Real time robust L1 tracker using accelerated proximal gradient approach. In: CVPR, pp. 1830–1837 (2012)
8. Liu, B., Huang, J., Yang, L., Kulikowsk, C.: Robust tracking using local sparse appearance model and k-selection. In: CVPR, pp. 1313–1320 (2011)
9. Zhong, W., Lu, H., Yang, M.: Robust object tracking via sparsity-based collaborative model. In: CVPR, pp. 1–8 (2012)
10. Jia, X., Lu, H., Yang, M.H.: Visual tracking via adaptive structural local sparse appearance model. In: CVPR, pp. 1822–1829 (2012)
11. Zhang, X., Li, W., Hu, W., et al.: Block covariance based L1 tracker with a subtle template dictionary. Pattern Recogn. **46**(7), 1750–1761 (2013). (Special Issue: Sparse representation for event recognition in video surveillance)
12. Zhang, T., Ghanem, B., Liu, S., Ahuja, N.: Robust visual tracking via structured multi-task sparse learning. IJCV **101**(2), 367–383 (2013)
13. Li, H., Shen, C., Shi, Q.: Real-time visual tracking with compressed sensing. In: CVPR, pp. 1305–1312 (2011)
14. Zhang, S., Yao, H., Sun, X., Lu, X.: Sparse coding based visual tracking: review and experimental comparison. Pattern Recogn. **46**(7), 1772–1788 (2013)
15. Lu, W., Okuma, K., Little, J.: Tracking and recognizing actions of multiple hockey players using the boosted particle filter. Image Vis. Comput. **27**, 189–205 (2009)
16. Breitenstein, M., Reichlin, F., Leibe, B., Koller-Meier, E., Gool, L.V.: Online multiperson tracking-by-detection from a single, uncalibrated camera. IEEE Trans. PAMI **33**(9), 1820–1833 (2011)
17. Yang, B., Huang, C., Nevatia, R.: Learning affinities and dependencies for multi-target tracking using a CRF model. In: CVPR, pp. 1233–1240 (2011)
18. Kuo, C., Huang, C., Nevatia, R.: Multi-target tracking by on-line learned discriminative appearance models. In: CVPR, pp. 685–692 (2010)
19. Berclaz, J., Fleuret, F., Turetken, E., Fua, P.: Multiple object tracking using k-shortest paths optimization. IEEE Trans. PAMI **33**(9), 1806–1819 (2011)
20. Huang, C., Wu, B., Nevatia, R.: Robust object tracking by hierarchical association of detection responses. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 788–801. Springer, Heidelberg (2008)
21. Jiang, H., Fels, S., Little, J.: A linear programming approach for multiple object tracking. In: CVPR, pp. 1–8 (2007)
22. Kuo, C.H., Nevatia, R.: How does person identity recognition help multi-person tracking? In: CVPR, pp. 1217–1224 (2011)

23. Milan, A., Roth, S., Schindler, K.: Continuous energy minimization for multitarget tracking. IEEE Trans. PAMI **36**(1), 58–72 (2014)
24. Pirsiavash, H., Ramanan, D., Fowlkes, C.: Globally-optimal greedy algorithms for tracking a variable number of objects. In: CVPR, pp. 1201–1208 (2011)
25. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. IEEE Trans. PAMI **34**(7), 1409–1422 (2012)
26. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV **60**(2), 91–110 (2004)
27. Szummer, M., Kohli, P., Hoiem, D.: Learning CRFs using graph cuts. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 582–595. Springer, Heidelberg (2008)
28. Yang, J., Yang, M.H.: Top-down visual saliency via joint CRF and dictionary learning. In: CVPR, pp. 2296–2303 (2012)
29. Lozano, A.C., Swirszcz, G., Abe, N.: Group orthogonal matching pursuit for variable selection and prediction. In: NIPS (2009)
30. Yang, J., Yu, K., Huang, T.: Supervised translation-invariant sparse coding. In: CVPR (2010)
31. Cheng, X., Li, N., Zhang, S., et al.: Robust visual tracking with SIFT features and fragments based on particle swarm optimization. Circuits Syst. Signal Process. **33**(5), 1507–1526 (2014)
32. Yoon, J.H., Kim, D.Y., Yoon, K.-J.: Visual tracking via adaptive tracker selection with multiple features. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part IV. LNCS, vol. 7575, pp. 28–41. Springer, Heidelberg (2012)
33. Wang, X., Türetken, E., Fleuret, F., Fua, P.: Tracking interacting objects optimally using integer programming. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part I. LNCS, vol. 8689, pp. 17–32. Springer, Heidelberg (2014)
34. Joachims, T., Finley, T., Yu, C.: Cutting-plane training of structural SVM. Mach. Learn. **77** (1), 27–59 (2009)

A Wireless Sensor Network Application with Distributed Processing in the Compressed Domain

Mauricio González, Javier Schandy, Nicolás Wainstein, Martín Bertrán, Natalia Martínez, Leonardo Barboni, and Alvaro Gómez(✉)

Facultad de Ingeniería, Instituto de Ingeniería Eléctrica, Universidad de la República, Montevideo, Uruguay
{mgonzalez,jschandy,nwainstein,mbertran, nataliam,lbarboni,agomez}@fing.edu.uy

Abstract. Wireless Sensor Networks are being used in multiple applications and they are becoming popular particularly in precision-agriculture and environmental monitoring. Their low-cost enables to build distributed deployments with large spatial density of nodes. They have been traditionally used to build maps describing scalar fields varying in time and space. However, in the recent years, image capturing capable nodes have appeared allowing to measure more complex data but imposing new challenges for the processor and memory constrained nodes.

Transmission of large images over a Wireless Sensor Network is a costly operation since most of the power consumption at the node is due to the operation of its radio. Hence, it is desirable to process and extract interesting features from the images at the node in order to transmit the important information and not all the images. However, image processing is also complicated by low processor and memory resources at the node. An image is usually delivered in JPEG format by the node's camera and stored in flash memory but, with current typical node configurations, memory resources are insufficient to open the image file and perform the image processing algorithms on the pixels of the image. To overcome this limitation, image processing can be done in the compressed domain parsing the JPEG file and working directly on the Discrete Cosine Transform coefficients of the compressed image blocks as soon as they are decoded. In this article, we present an agricultural Wireless Sensor Network application that implements block based classification in the compressed domain. In this application, image-sensor nodes are placed on insect pest traps to quantify pest population in fruit trees.

Keywords: Wireless sensor network · Pest monitoring · Compressed domain · Block based classifier · JPEG · DCT

The authors are grateful for the support of CSIC-Universidad de la República and INIA-FPTA Research Project 313.

P.L. Mazzeo et al. (Eds.): AMMDS 2014, pp. 104–115, 2014.
DOI: 10.1007/978-3-319-13323-2_9

1 Introduction

Wireless Sensor Networks (WSN) are comprised of small programmable devices named nodes. Each node is composed of a micro-controller, sensors and a radio to communicate wirelessly with neighboring nodes. Their increasingly widespread usage is due to the following reasons: (i) the node low-cost enables to build distributed deployments easily scalable with large spatial density of nodes per unit area with reduced cost of installation and maintenance, (ii) these deployments enable to build maps describing scalar fields varying in time and space (e.g. temperature, humidity), (iii) the nodes operate with low current consumption, so that they can achieve several months of battery lifetime.

Most WSN applications have been traditionally restricted to scalar measuring nodes, see for example [10,14], but recently image capturing capable nodes are being integrated with the challenge of handling more complex data over the networks. With this new scenario, image processing at the node becomes important to reduce radio transmissions in order to keep the current consumption bounded and hence achieve long battery lifetime. Some works in this line are presented in [6,12].

WSNs have received considerable research and development effort in order to enhance their capabilities to be used for precision agriculture and environmental monitoring. In these areas, WSNs enable to manage the farm productivity, allowing product quality enhancement with reduced operational cost.

Image processing in the compressed domain refers to a wide range of algorithms that can be performed directly on compressed images with no prior decompression or with only partial decompression. Research in this field started in the nineties (see for example [3,15]) but the increase in power and memory resources of computers left this kind of processing almost unnecessary for most applications. With plentiful resources, compressed images can be firstly decompressed in order to apply algorithms for the usual spatial domain. However, with restricted processing and memory resources, processing in the compressed domain may make sense constituting a powerful tool. WSNs are one of these applications with low computing and memory resources that can benefit from compressed domain techniques. WSNs are not designed to transmit large amounts of data and transmitting images implies having to modify the existing network protocols to increase the performance of the system. Having the possibility to process efficiently an image at the node and transmit only interesting information, not only enhances the battery life time by reducing the amount of data transmitted (resulting in decreasing the time the radio is in active mode), but also widely simplifies the network protocols as the problem turns into a classical WSN application, where few bytes are transmitted periodically.

In WSNs, cameras that can be attached to a wireless node usually have an integrated JPEG compression engine and deliver a JPEG file. Accessing the image file stream and processing in the compressed domain is a task that can be performed efficiently in the node even with low resources.

In this article, we present an agricultural WSN application that implements block based classification in the compressed domain. The work is part of an

ongoing project between our University and fruit producers that looks forward to deploying a distributed pest monitoring system. As a first step in this project, an image-sensor node is being designed capable of periodically taking an image of the inside of a pest trap, analyze the image and deliver information via the radio channel. Some previous related work with image capable WSNs in agriculture can be found in literature such as [9,16], but, up to our knowledge, the approach of block based classification in the compressed domain at the node has not been implemented before.

Following this introduction, Sect. 2 presents the main concepts of processing in the compressed domain and introduces block based classification. Section 3 introduces the pest monitoring WSN application, the implementation of the image node and the compressed domain processing at the node. Results of block based classification are shown on artificial and real images of trapped insects. Finally Sect. 4 presents some concluding remarks.

2 Processing in the Compressed Domain

2.1 JPEG Compressed Images

JPEG is a standard for lossy compression and codification of images. In the basic version of the standard, the image is tessellated in 8×8 blocks and each block is transformed from the spatial domain to the frequency domain using the Discrete Cosine Transform (DCT). For each block, the first DCT coefficient is called the DC coefficient and it is equivalent to the average intensity value of the block, the rest are known as AC coefficients associated to the other frequency components present in the block. The DCT coefficients of each block undergo quantization using different weights adapted to the different frequency components according to their perceptual importance.

This process of quantization is the step that introduces loss since the DCT coefficients are divided by their respective quantization weights and the result is rounded off. After quantization, the remaining non zero DCT coefficients of an 8×8 block are ordered following a zig-zag pattern and subsequently zero run length encoded and codified using Huffman variable length coding. When the image has colors they are represented in the luminance-chrominance YCbCr space. The Cb-Cr components have less bandwidth than the luminance so they can be further downsampled. These components can be downsampled in different proportions, for example 4:1:1 which means that for every 4 luminance blocks there is only 1 Cb and 1 Cr block. A group of such luminance-chrominance blocks (4 Y, 1 Cb and 1 Cr in the example) form a Minimum Coded Unit (MCU). The resulting JPEG file consists of a header with image information and the applied quantization and Huffman tables, followed by the variable length coded blocks. A complete explanation of the JPEG standard can be found in [17]. Figure 1 presents a diagram of the compression process in JPEG.

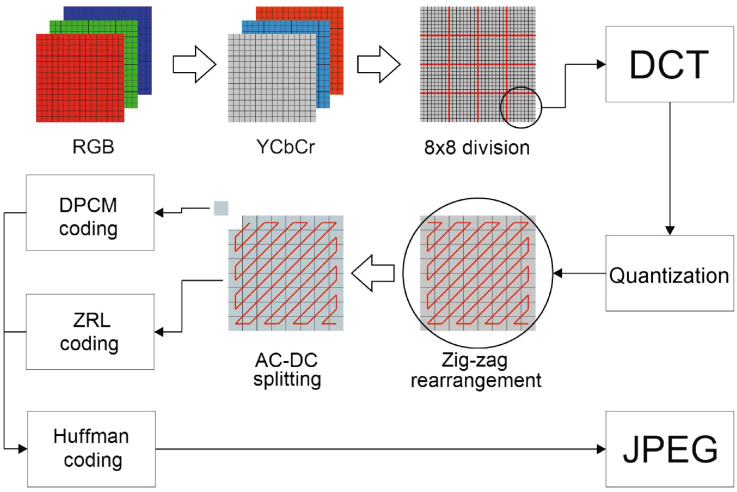


Fig. 1. Basic block diagram of the JPEG standard.

2.2 Compressed Domain Processing

The decompression of a JPEG file follows the inverse sequence of steps. Image processing in the compressed domain can be performed at different stages of the decompression process. In increasing computational complexity, the data that can be used from the JPEG file is:

- Coding length of each block:
  The coding length of a block is the number of bits that are used to represent the block in the JPEG file. This value is a measure of the complexity of the block as it is related to the entropy of the block.
- Decoded DC component of each block:
  Only the first DCT component of each block is kept while decoding. The DC component is the mean value of the block.
- Decoded DCT components of each block:
  All the DCT components of the blocks are decoded, the DC and the AC components. The coefficients can be used individually or combined as different measures of the energy of the block at several frequency sub-bands.
- Inverse DCT transformed values of selected blocks:
  The pixel values of a block can be obtained by inverse transforming the DCT coefficients if needed for some selected blocks.

Processing in the compressed domain can be performed using some or all of this data, according to the selected algorithm. Apart from the type of data, processor and memory requirements of an algorithm are less when it operates sequentially one block at a time. Requirements rise when the algorithm uses information of all the blocks and/or spatial relationships between the blocks

which implies more than one traverse of the JPEG file or memory allocation for temporary data.

Since it is possible to recover the DCT coefficients, and the DCT transform is linear, every linear operation on the spatial domain can be ported to the compressed domain. Also some non linear methods can also be ported to the compressed domain. Examples of the algorithms used in the compressed domain are: (i) Arithmetic pixelwise operations [15], (ii) Contrast enhancement [7], (iii) Linear Filtering [18]. A more extensive compendium of image and video processing in the compressed domain is presented in [11].

Most relevant to this article, block based classification can be implemented efficiently in the compressed domain since good features to describe the contents of a block can be easily computed from the DCT coefficients. Several classical supervised classifiers can be trained offline with powerful processing resources and used afterwards as block based classifier with low processing requirements.

2.3 Block Based Classifier

Block based classification is an operation that can be performed block by block as soon as they are decoded. Once decoded, the 8×8 blocks are then classified as a unit, allowing for image segmentation, and/or selective transmission of blocks that contain relevant information.

An example of a simple, low computing cost supervised classifier can be implemented using Fisher's linear discriminant analysis (LDA) [4]. The classifier can use easy to compute features on the DCT coefficients of the blocks. This method finds the direction in which the feature vectors can be projected that gives the best separability between classes. The training of the classifier can be done offline, thus limiting the required online processing to calculating the features, applying a dot product with the projection vector, and classifying via a threshold.

Figure 2 shows some results on the "17 Category flower dataset" from the Visual Geometry Group (U.of Oxford) [13] (hereinafter referred to as Flowers dataset) and the "Airplane dataset" provided by Caltech Vision Group [1] (hereinafter referred to as "Airplanes" dataset). A block based classifier was trained with 450 images and tested on 250 images from the Flowers dataset. These images were randomly chosen from those that had a ground truth to compare to. A classifier was also trained for the "Airplanes" dataset In this case, 267 images were selected from those that had the sky as background and the set was randomly divided in 157 images for the training set and the remaining 110 for the test set. The features chosen on the DCT coefficients were the DC component, total AC energy of the block, relative energy of the first, second and third bands (15 features, 5 features for each image channel Y, Cb an Cr).

Considering $D = (d_0, d_1, ..., d_{63})^t$ the vector of DCT coefficients of a block in zig-zag order for one of the Y,Cb,Cr channels the features are computed as:

$$\left(d_0, \sum_{i=1}^{63} {d_i}^2, \frac{\sum_{i=1}^{2} {d_i}^2}{\sum_{j=1}^{63} {d_j}^2}, \frac{\sum_{i=3}^{5} {d_i}^2}{\sum_{j=1}^{63} {d_j}^2}, \frac{\sum_{i=6}^{9} {d_i}^2}{\sum_{j=1}^{63} {d_j}^2} \right) \tag{1}$$

Table 1. Confusion matrices at an operating point with a classifier threshold set to 1.5

		Predicted positive	Predicted negative
Flowers	Actual positive	83.83 %	14.75 %
	Actual negative	16.17 %	85.25 %
Airplanes	Actual positive	85.17 %	6.94 %
	Actual negative	14.83 %	93.06 %

Figure 2 shows the results on some images of the databases. Table 1 presents the confusion matrices for the performance on the training sets of both datasets at an operating point with a classifier threshold of 1.5.

Fig. 2. Results of block based classification. Please refer to text for complete explanation.

The classifier seems adequate for extracting the objects of interests from the background. Thus, the desired objective of selective compression can be easily met without much loss in relevant information. If only the blocks corresponding to detected objects were to be transmitted, analyzed or stored, the "interesting blocks" to whole image blocks ratio would be 42 % for "Flowers" and 13 % for "Airplanes". Note that these numbers depend largely on the proportion of "interesting blocks" to background in the image and greater efficiencies are achieved in sparsely populated images.

3 The WSN Pest Monitoring Application

Processing in the compressed domain is applied to a WSN for monitoring pest population in fruit trees. In this application, images of pest traps are acquired and analyzed to control pest population on a daily basis. Images could be transmitted over the network to be collected and analyzed on a server but this imposes large traffic in the network that implies complex network protocols and excessive energy consumption at the nodes. Reduction of transmission payload is mandatory to achieve a simple network and low energy consumption of the nodes. With this objective, block based classification is a useful tool to detect the number of insects in the trap and transmit one radio packet with this information instead of transmitting hundreds of packets with the complete image.

3.1 Application Description

The lepidopterous insect pest (moths) produces diseases in trees. The moths lay eggs from which larvae are born and they produce lesions to the fruit. The control of the pest population is implemented by means of using plastic traps with a sticky bottom side and pheromone lures. The trap can capture male adult moths attracted by the female pheromone lures. A person, who periodically travels through crops, is in charge of performing the counting of insects caught in the trap and eventually clean trap bottoms of pest crowded traps.

In an ongoing project between the University and fruit producers, a wireless image-sensor node is being designed capable of taking images inside the trap, analyze the images at the node and transmit the important information via radio channel. The final pest monitoring system will: (i) enable simple pest monitoring of large areas, (ii) simplify the maintenance of the traps since the person in charge will only be required for trap cleaning when needed, (iii) enable early alerts in case of pest infection thus allowing performing localized fumigation in the crop with reduced pesticide usage and hence avoiding environmental and water pollution.

3.2 The Wireless Image-Sensor Node (WimSN)

The designed WimSN is based on the CM3000 [5] node that features the TI MSP430F1611 16-bit RISC microprocessor with a program memory flash of 48 kB, data RAM of 10 kB and an external flash of 1 MB. The wireless communication is implemented by means of the RF Chip TI CC2420 (IEEE 802.15.4 2.4 GHz standard compliant). The selected camera module is the LinkSprite JPEG Color Camera TTL Interface-Infrared LS-Y201 [8]. It requires DC 3.3 V voltage power supply and the current consumption is around 80 mA. The camera module can capture VGA/QVGA and lower resolution images, it integrates a JPEG compression engine, serial communication and 60 degrees viewing angle lens.

The WimSN uses Contiki OS [2], an Event-Driven real time operating system (RTOS) oriented to WSN applications in constrained hardware. This RTOS

manages the hardware resources and includes different libraries such as network stacks and a file system. Contiki OS scheduler manages sleep modes powering down the microprocessor when there is neither processing needed nor events scheduled in the event queue.

Figure 3 shows the implemented block diagram.

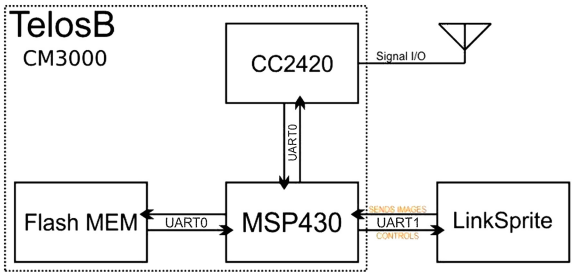


Fig. 3. Block diagram of the implemented wireless image-sensor node.

The WimSN is enclosed in a watertight compartment and it is attached to an acrylic delta shape trap. The trap bottom (which collects the trapped insects and is photographed by the WimSN camera) hangs from the delta trap and can be easily dismounted for cleaning or inspection. Figure 4 shows the 3D design and the implemented device. Figure 5 presents images acquired with the WimSN in the designed trap.

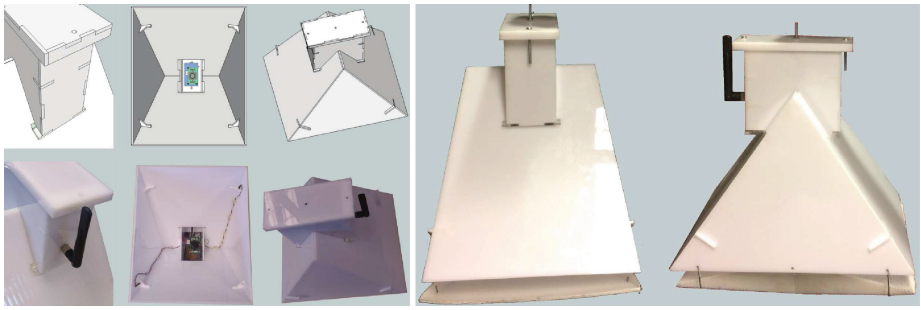


Fig. 4. The WimSN integrated to the pest trap. Left: 3D design and implemented device showing the exterior and interior of the trap. Right: the trap with the hanging bottom.

3.3 Block Based Classification

Implementation at the Node. As explained in Sect. 3.2, the microcontroller receives a JPEG compressed image from the camera and stores the image in flash memory. In order to apply LDA, the JPEG file has to be parsed and the DCT

blocks have to be decoded from the file stream but no further decompression is needed. A full JPEG decompression is not only impractical but even impossible due to the reduced hardware capabilities[1], so authors used a custom minimal decoder. The implemented decoder does not perform a full JPEG decompression but obtains the DCT coefficients of each Y, Cb and Cr block per block.

A block based classifier based on LDA was implemented as explained in Sect. 2.3. In the case of the WimSN, the implemented classifier uses only features from the Y luminance channel (DC component, total AC energy of the block, relative energy of the first, second and third bands).

After acquiring an image, the JPEG file is parsed and each block is classified as soon as it is decoded. The classification indicates if a block is part of an insect or is part of the background and the coordinates of the positive blocks are stored in a list that can be transmitted afterwards.

Optionally, under the hypothesis that the moths are not overlapped, a customized labeling algorithm can be applied in order to reduce the amount of stored data. The size of a moth in the image is bigger than an 8×8 block, so one moth is detected in several blocks. Taking advantage of the fact that DCT blocks are decoded row by row of the image, when an horizontal burst of positive classified blocks is detected, only the coordinates of the middle block is stored. After the classification of all the blocks, the list is inspected to identify and merge vertically connected regions. Therefore, the system is capable of establishing the coordinates near the centroid of each insect.

Results. In the current stage of the project, trap images are still insufficient in order to train and test thoroughly the classification of insects. The classification algorithm implemented at the node is tested on a database augmented with images of artificial insects. To build this database, images were acquired with insects simulated with objects of similar shape and taking into account the variability of other aspects (variable illumination, dirt at the trap bottom, etc.). Figure 5 shows images of the trap bottom with real and simulated insects acquired with the WimSN.

Figure 6 shows an image acquired in the designed trap and the result of block based classification.

The training and testing sets of 8×8 blocks are built by manual segmentation of the insects in the images. A block in an image is considered an insect block if at least the 80 % of it is covered by a segmented insect. A dataset of 28 images (134400 8×8 blocks) was considered for testing and training. Although not extensive to do a complete train/test performance analysis, the primary results are encouraging. Considering training sets of 22 images (105600 blocks) and testing sets of 6 images (28800 blocks) the system shows an area under the curve (AUC) of the Receiver operating characteristic (ROC) of over 0.85.

[1] Note that the included flash memory would be almost filled completely with a full RAW image leaving little space for other data (considering that a 3 channel, 640×480 file occupies more than 900 kB).

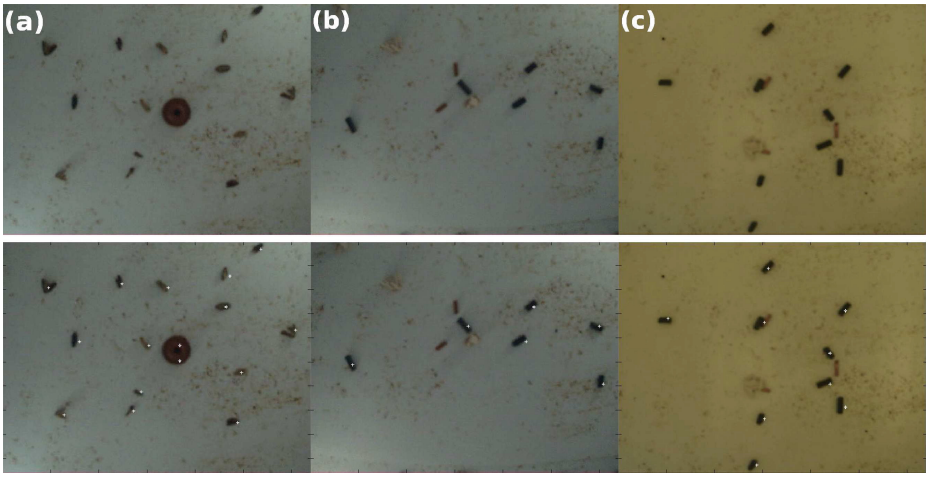


Fig. 5. Above, images acquired with the designed WimSN node. Below, same images with detected insects. (a) Image containing real insects. The round object is the pheromone lure. (b, c) Image of simulated insects. Images (a) and (b) were acquired with the trap self illumination, while image (c) was acquired with the trap in daylight.

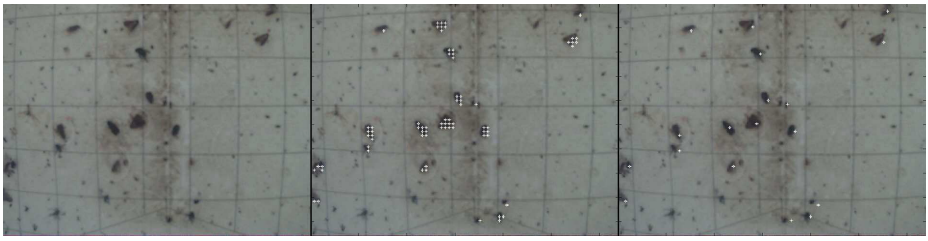


Fig. 6. (a) Image acquired with the designed WimSN and trap. (b) Position of the positive blocks detected with the block based classifier. (c) Position of the connected components detected with the optional labelling algorithm

3.4 Impact of Distributed Processing on the WSN

The impact of distributed processing in our application is twofold.

1. WSN were not designed to transmit large amounts of data. The transmission of images implies having to modify the existing network protocols to increase the performance of the system (e.g. enhance synchronization strategies over the radio channel). Having the possibility to transmit only the information of the detected insects in a couple of radio packets widely simplifies the network protocol selection as the problem turns into a classical WSN application, where few bytes are transmitted periodically. In this scenario, well established network protocols and applications already implemented in the real time operating system can be used to collect data in tree or mesh topology networks.

2. Transmitting only the information of the detected insects also enhances the battery lifetime of the nodes of the WSN. Sending JPEG images represents the 7.3 % of the consumption of battery power of the WimSN node. This could be acceptable in unicast single-hop transmissions but not for large covering WSNs with tree or mesh multi-hop topologies. For example, for a branch network with 7 hops, the node closest to the sink[2] shall transmit its own image but also receive and transmit 6 images of its' children nodes. This implies that the energy consumed by the radio activity would be increased considerably in this node degrading its energy independence.

   On the other hand, the implemented detection algorithm represents the 1.2 % of the consumption of battery power of the WimSN but retransmissions represent only 0.1 % of the energy consumption at the node. In this configuration, the estimated consumption of the battery charge is 2.62 mAh per day (acquiring and processing two images per day) which means an energy autonomy of over 30 months with typical AA batteries. Hence, with distributed processing, the energy autonomy of the nodes close to the sink is no longer an issue.

4 Concluding Remarks

Algorithms for image processing and pattern recognition in the compressed domain are useful tools for applications with low computational resources and strict energy consumption requirements. WSNs with imaging nodes are one of this kind of applications that can benefit from compressed domain techniques.

A wireless sensor node was designed and assembled for a WSN dedicated to distributed pest monitoring on fruit trees. The node implements a block based classifier in the compressed domain. In this case a simple LDA classifier was used but the experience can be easily extended to other supervised classifiers in the future. The preliminary results of the classification are promising. Although, further experimentation and a bigger image dataset is necessary to train/test the system, this step in our project has shown that the approach is valid and it enables: (i) battery lifetime enhancement, and (ii) network simplification.

The next step in our project is the deployment of some WimSN nodes in the fruit plantation in order to capture more images during the fruit growing season. In this first deployment, nodes will be working in a dual mode transmitting the images and also the local classification. Classification will be evaluated and the acquired images will allow to build an important database that will enable to incrementally retrain the classifier.

Our future work will include the evaluation of classification on the image database acquired at the plantation. That evaluation will tell us if this simple approach is sufficient or if other classifiers/algorithms are required at the nodes. Anyway, the framework for compressed domain classifiers/algorithms is already set and easy to extend.

[2] The sink node receives all the information from the network. This node is usually attached to a computer and does not have energy restrictions.

References

1. Caltech. Computational Vision Group - Archive. http://www.vision.caltech.edu/html-files/archive.html
2. The Contiki Community. Contiki - The Open Source OS for the Internet of Things.
3. de Queiroz, R.L.: Processing JPEG-compressed images and documents. IEEE Trans. Image Process. **7**(12), 1661–1672 (1998)
4. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley, New York (2012)
5. Maxfor Technology Inc., Maxfor Digital Brochure (2011)
6. Karlsson, J.: Image compression for wireless sensor networks. Master thessis in computing Science (2007)
7. Lee, S.: An efficient content-based image enhancement in the compressed domain using retinex theory. IEEE Trans. Circ. Syst. Video Technol. **17**(2), 199–213 (2007)
8. Linksprite. LinkSprite JPEG Color Camera Serial UART Interface (2012)
9. Lloret, J., Bosch, I., Sendra, S., Serrano, A.: A wireless sensor network for vineyard monitoring that uses image processing. Sensors **11**, 6165–6196 (2011)
10. López, J.A., Soto, F., Sánchez, P., Iborra, A., Suardiaz, J., Vera, J.A.: Development of a sensor node for precision horticulture. Sensors **9**(5), 3240–3255 (2009)
11. Mukhopadhyay, J.: Image and Video Processing in the Compressed Domain. CRC Press, Boca Raton (2011)
12. Nikolakopoulos, G., Kandris, D., Tzes, A.: Adaptive compression of slowly varying images transmitted over wireless sensor networks. Sensors **10**(8), 7170–7191 (2010)
13. Nilsback, M.-E., Zisserman, A.: A visual vocabulary for flower classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 1447–1454 (2006)
14. Pierce, F.J., Elliott, T.V.: Regional and on-farm wireless sensor networks for agricultural systems in eastern washington. Comput. Electron. Agric. **61**(1), 32–43 (2008)
15. Shen, B., Sethi, I.K., Bhaskaran, V.: DCT domain alpha blending. In: 1998 Proceedings of International Conference on Image Processing, ICIP 98, vol. 1, pp. 857–861. IEEE (1998)
16. Tirelli, P., Borghese, N.A., Pedersini, F., Galassi, G., Oberti, R.: Automatic monitoring of pest insects traps by Zigbee-based wireless networking of image sensors. In: I2MTC-IEEE, pp. 1–5 (2011)
17. Gregory, K.: Wallace. the JPEG still picture compression standard. Commun. ACM **34**(4), 30–44 (1991)
18. Zhuang, L., Zhao, R., Yu, N., Liu, B.: SVD based linear filtering in DCT domain. In: 2010 17th IEEE International Conference on Image Processing (ICIP), pp. 2769–2772. IEEE (2010)

Author Index