ADVANCES IN INTELLIGENT SYSTEMS AND COMPUTING 225

Dhinaharan Nagamalai
Ashok Kumar
Annamalai Annamalai (Eds.)

Advances in Computational Science, Engineering and Information Technology

Proceedings of the Third International Conference on Computational Science, Engineering and Information Technology (CCSEIT-2013), KTO Karatay University, June 7–9, 2013, Konya, Turkey – Volume 1

Springer

Advances in Intelligent Systems and Computing

Volume 225

Series Editor

J. Kacprzyk, Warsaw, Poland

For further volumes:
http://www.springer.com/series/11156

Dhinaharan Nagamalai · Ashok Kumar
Annamalai Annamalai
Editors

Advances in Computational Science, Engineering and Information Technology

Proceedings of the Third International Conference on Computational Science, Engineering and Information Technology (CCSEIT-2013), KTO Karatay University, June 7–9, 2013, Konya, Turkey – Volume 1

Springer

Editors
Dhinaharan Nagamalai
Dept. of Computer Engineering
Faculty of Engineering
KTO Karatay University
Konya
Turkey

Annamalai Annamalai
Dept. of Electrical and Computer Engineering
Prairie View A & M University
Texas
USA

Ashok Kumar
School of Computing and Informatics
University of Louisiana at Lafayett
Louisiana
USA

ISSN 2194-5357 ISSN 2194-5365 (electronic)
ISBN 978-3-319-00950-6 ISBN 978-3-319-00951-3 (eBook)
DOI 10.1007/978-3-319-00951-3
Springer Cham Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013939955

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The Third International Conference on Computational Science, Engineering and Information Technology (CCSEIT-2013) was held in Konya, Turkey, June 7–9, 2013. CCSEIT-2013 attracted many local and international delegates, presenting a balanced mixture of intellect from the East and from the West. The goal of this conference series is to bring together researchers and practitioners from academia and industry and share cutting-edge development in the field. The conference will provide an excellent international forum for sharing knowledge and results in theory, methodology and applications of Computational Science, Engineering and Information Technology. Authors were invited to contribute to the conference by submitting articles that illustrate research results, projects, survey work and industrial experiences describing significant advances in various areas of Computational Science, Engineering and Information Technology.

The CCSEIT-2013 Committees rigorously invited submissions for many months from researchers, scientists, engineers, students and practitioners related to the relevant themes and tracks of the conference. This effort guaranteed submissions from an unparalleled number of internationally recognized top-level researchers. All the submissions underwent a strenuous peer-review process which comprised expert reviewers. These reviewers were selected from a talented pool of Technical Committee members and external reviewers on the basis of their expertise. The papers were then reviewed based on their contributions, technical content, originality and clarity. The entire process, which includes the submission, review and acceptance processes, was done electronically. All these efforts undertaken by the Organizing and Technical Committees led to an exciting, rich and a high quality technical conference program, which featured high-impact presentations for all attendees to enjoy, appreciate and expand their expertise in the latest developments in computer network and communications research.

In closing, CCSEIT-2013 brought together researchers, scientists, engineers, students and practitioners to exchange and share their experiences, new ideas and research results in all aspects of the main workshop themes and tracks, and to discuss the practical challenges encountered and the solutions adopted. We would like to thank the General and Program Chairs, organization staff, the members of the Technical Program Committees and external reviewers for their excellent and tireless work. We sincerely wish that all attendees benefited scientifically from the conference and wish them every success in their research.

It is the humble wish of the conference organizers that the professional dialogue among the researchers, scientists, engineers, students and educators continues beyond the event and that the friendships and collaborations forged will linger and prosper for many years to come. We hope that you will benefit from the fine papers from the CCSEIT-2013 conference that are in this volume and will join us at the next CCSEIT conference.

Dhinaharan Nagamalai
Ashok Kumar
Annamalai Annamalai

Organization

General Chair

Ali Okatan	KTO Karatay University, Turkey
David C. Wyld	Southeastern Louisiana University, USA

General Co-Chairs

Jae Kwang Lee	Hannam University, South Korea
Michal Wozniak	Wroclaw University of Technology, Poland

Steering Committee

Natarajan Meghanathan	Jackson State University, USA
Chih-Lin Hu	National Central University, Taiwan
Dhinaharan Nagamalai	KTO Karatay University, Turkey
Chin-Chih Chang	Chung Hua University, Taiwan
Ashok Kumar	University of Louisiana at Lafayette, USA

Volume Editor

Annamalai A	Prairie View A & M University, USA

Publicity Chairs

Kasym Oztoprak	KTO Karatay University, Konya, Turkey
Ali Ozturk	KTO Karatay University, Konya, Turkey
Refik Caglar Kyzylyrmak	KTO Karatay University, Konya, Turkey

Program Committee Members

Piotr Zwierzykowski	Poznan University of Technology, Poland
Luca Caviglione	National Research Council of Italy (CNR) - ISSIA
Danda B. Rawat	Eastern Kentucky University, USA
Gregorio Martinez	University of Murcia, Spain
Antonio Ruiz-Martinez	University of Murcia, Spain
Emilio Jimenez Macias	University of La Rioja, Spain
Eugenia Moreira Bernardino	School of Technology and Management, Polytechnic Institute of Leiria, Portugal
Kamran Arshad	University of Greenwich, United Kingdom
Josip Lorincz	FESB, University of Split, Croatia
Carlos Miguel Tavares Calafate	Universidad Politecnica de Valencia, Spain
Namje Park	Jeju National University, Korea
Praveen Ranjan Srivastava	BITS Pilani, Rajasthan, India
Chitra	PSG College of Technology-Coimbatore, India
A.Velumani	Avinashilingam University-Coimbatore, India
Aarti M. Karande	Mumbai University, India
Abdul Aziz	University of Central Punjab, Pakistan
Abdul Kadir Ozcan	KTO Karatay University, Turkey
Acharyulu K.V.L.N	Bapthala Engineering College, India
Aditi Sharan JNU	New Delhi, India
Ajay K Sharma	Dr B R Ambedkar National Institute of Technology, India
Amandeep Singh Thethi	Guru Nanak Dev University Amritsar, India
Amit Kumar Singh Sanger	Hindustan College of Science and Technology, India
Andreas Riener	Johannes Kepler University Linz, Austria
Andy Seddon	Asia Pacific Institute of Information Technology, Malaysia
Ankit Chaudhary	BITS Pilani, India
Antelin vijila	Manonmaniam Sundaranar University, India
Arokiasamy A	Eastern Mediterranean University, Cyprus
Arunadevi J	Thiagarajar college, Madurai, India
Arvinth kumar	M.S. University, India
Ashok kumar Sharma	YMCA Institute of Engineering, India
Athanasios Vasilakos	University of Western Macedonia, Greece
Atilla Elci	Eastern Mediterranean University, Cyprus
Azween Bin Abdullah	Universiti Teknologi Petronas, Malaysia
Kalpana	Avinashilingam University-Coimbatore, India

Sarojini	Avinashilingam University-Coimbatore, India
Bai Li	IT Technical Analyst, Australia
Balakrishnan G	Indra Ganesan College of Engineering, India
Balasubramaniam	Manonmaniam Sundaranar University, India
Bhadka H.B	C.U. Shah College, India
Bharat Bhushan Agarwal	I.F.T.M., University, India
Bhupendra Gupta	Indian Institute of Information Technology Design & Manufacturing Jabalpur, India
Binod Kumar Pattanayak	Siksha O Anusandhan University, India
Bong-Han Kim	Chongju University, South Korea
Boo-Hyung Lee	KongJu National University, South Korea
Brajesh Kumar Kaushik	Indian Institute of Technology, India
Meena	Avinashilingam University-Coimbatore, India
Charalampos Z. Patrikakis	National Technical University of Athens, Greece
Chih-Lin Hu	National Central University, Taiwan
Chin-Chih Chang	Chung-Hua University, Taiwan
Cho Han Jin	Far East University, South Korea
Choudhari	Bhagwati Chaturvedi College of Engineering, India
Cynthia Dhinakaran	Hannam University, South Korea
Cynthia Mary	Kumaraguru College of Technology, India
Danda B Rawat	Old Dominion University, USA
David W Deeds	Shingu College, South Korea
Debasis Giri	Haldia Institute of Technology, India
Deepak garg	Thapar University, India
Dhinaharan Nagamalai	Wireilla Net Solutions PTY LTD, Australia
Dimitris Kotzinos	Technical Educational Institution of Serres, Greece
Dinakaran M	VIT University, Vellore
Dong Seong Kim	Duke University, USA
E.P. Sumesh	Amrita University-Coimbatore, India
El boukhari Mohamed	University Mohamed First, Morocco
Emmanuel Bouix	iKlax Media, France
Eric Renault	Institut Telecom – Telecom SudParis, France
Farhat Anwar	International Islamic University, Malaysia
Firkhan Ali Bin Hamid Ali	Universiti Tun Hussein Onn Malaysia, Malaysia
Ford Lumban Gaol	Bina Nusantara University, Indonesia
G. Jayagouri	Avinashilingam University-Coimbatore, India
G.Padmavathi	Avinashilingam University-Coimbatore, India
Geuk Lee	Hannam University, South Korea
Girija Chetty	University of Canberra, Australia
Gomathy C	Sathyabama University, India

Hamid Reza Karimi	University of Agder, Norway
Hao Shi	Victoria University, Australia
Hao-En Chueh	Yuanpei University, Taiwan, R.O.C.
Hariharan S	B.S.Abdur Rahman University, India
Hariharan S	Pavendar Bharathidasan College of Engineering and Technology, India
He Xu	Nanjing University of Post and Telecommunications, China
Henrique J.A. Holanda	UERN - Universidade do Estado do Rio Grande do Norte
Henrique Joao Lopes Domingos	University of Lisbon, Portugal
Ho Dac Tu	Waseda University, Japan
Hoang	Huu Hanh, Hue University, Vietnam
Hwangjun Song	Pohang Univ of Science and Technology, South Korea
I. Elizabeth Shanthi	Avinashilingam University-Coimbatore, India
Indrajit Bhattacharya	Kalyani Govt. Engg. College, India
Jacques Demerjian	Communication & Systems, Homeland Security, France
Jae Kwang Lee	Hannam University, South Korea
Jalel Akaichi	University of Tunis, Tunisia
Jansirani	Manonmaniam Sundaranar University, India
Jeong-Hyun Park	Electronics Telecommunication Research Institute, South Korea
Jitendra Prithviraj	GGITS, Jabalpur, India
Jivesh Govil	Cisco Systems Inc - CA, USA
Johann Groschdl	University of Bristol, UK
Johnson Kuruvila	Dalhousie University, Halifax Canada
Jose Enrique Armendariz-Inigo	Universidad Publica de Navarra, Spain
Jung-Gil Song	Hannam University, South Korea
Jungwook Song	Konkuk University, South Korea
Jyoti Singhai	Electronics and Communication Deptt-MANIT, India
K.Somasundaram	Gandhigram Rural University, India
K.Thangavel	Periyar University-Salem, India
Kamaljit I Lakhtaria	Atmiya Institute of Technology & Science, India
Kami Makki S	Lamar University, USA
Kannan A	K.L.N. College of Engineering, India
Kannan	Anna University, Chennai, India
Khamish Malhotra	University of Glamorgan, UK
Khoa N. Le	Griffith University, Australia
Krishnan N	Manonmaniam Sundaranar University, India

Krzysztof Walkowiak	Wroclaw University of Technology, Poland
L.Ganesan	Alagappa University, India
Lakshmi Rajamani	Osmania University, India
Lu s Veiga	Technical University of Lisbon, Portugal
Lu Yan	University Of Hertfordshire, UK
Lylia Abrouk	University of Burgundy, France
M.Mayilvaganam	PSG College of Arts and Science-Coimbatore, India
M.Thangaraj	Madurai Kamaraj University-Madurai, India
Madhan KS	Infosys Technologies Limited, India
Manik Gupta	LSI Research and Development, India
Maniza Hijab	Osmania University, India
Maode Ma	Nanyang Technological University, Singapore
Marco Roccetti	Universty of Bologna, Italy
Masaki Murakami	Okayama University, Japan
Michael Peterson	University of Hawaii at Hilo, United States
Mohsen Sharifi	Iran University of Science and Technology, Iran
Murali R	Dr. Ambedkar Institute of Technology, Bangalore
Murugan D	Manonmaniam Sundaranar University, India
Murugeswari	M.S.University, India
Muthu Selvi R	Thiagarajar College of Engineering, India
Muthulakshmi	M.S.University, India
N.Ganesan	Director Regional Institute of Cooperative Management-Bangalore, India
N.Valliammal	Avinashilingam University-Coimbatore, India
Nabendu Chaki	University of Calcutta, India
Neerajkumar	Madha Vaisnavi Devi University, India
Nicolas Sklavos	Technological Educational Institute of Patras, Greece
Nidaa Abdual Muhsin	Abbas- University of Babylon, Iraq
Nitiket N Mhala	B.D. College of Engineering - Sewagram, India
P. Narayanasamy	Anna University-Chennai, India
P. Ilango	VIT, Chennai, India
P. Subashini	Avinashilingam University-Coimbatore, India
Patrick Seeling	University of Wisconsin - Stevens Point, USA
Paul D. Manuel	Kuwait University, Kuwait
Phan Cong Vinh	London South Bank University, United Kingdom
Ponpit Wongthongtham	Curtin University of Technology, Australia
Prabu Dorairaj	NetApp Inc, India
Pravin P. Karde	HVPM's College of Engineering & Technology - Amravati, India
Premanand K. Kadbe	Vidya Pratishthan's College of Engineering, India

R. Brindha	Avinashilingam University-Coimbatore, India
R. Baskaran	Anna University - Chennai, India
R. Vijayabhanu	Avinashilingam University-Coimbatore, India
Rafiqul Zaman Khan	Aligarh Muslim University, India
Rajalakshmi	Manonmaniam Sundaranar University, India
Rajendra Akerkar	Technomathematics Research Foundation, India
Rajesh Kumar P	The Best International, Australia
Rajesh Manonmaniam	Sundaranar University, India
RajeshBawa	Punjabi University, India
Rajkumar Kannan	Bishop Heber College, India
Rakesh Kumar Mishra	Feroze Gandhi Institute of Engineering and Technology, India
Rakhesh Singh Kshetrimayum	Indian Institute of Technology-Guwahati, India
Ramakrishnan H.V	Dr.MGR University, India
Ramanathan L	VIT University, India
Ramayah Thurasamy	Universiti Sains Malaysia, Malaysia
Ranjani Parthasarathi	Anna University-Chennai, India
Ravichandran C.S	SSK College of Engineering and Technology, India
Reena Dadhich	Govt. Engg. College Ajmer, India
Reza Ebrahimi Atani	Guilan University, Iran
Rituparna Chaki	West Bengal University of Technology, India
Roberts Masillamani	Hindustan University, India
S. Sivakumari	Avinashilingam University-Coimbatore, India
S. Anantha Lakshmi	Avinashilingam University-Coimbatore, India
S. Arumugam	Nandha Engg college, India
S.N. Geethalakshmi	Avinashilingam University-Coimbatore, India
Sadasivam	Manonmaniam Sundaranar University, India
Sagarmay Deb	Central Queensland University, Australia
Sajid Hussain	Fisk University, United States
Salah S. Al-Majeed	University of Essex, United Kingdom
Salman Abdul Moiz	Centre for Development of Advanced Computing, India
Sanguthevar Rajasekaran	University of Connecticut, USA
Sarmistha Neogy	Jadavpur University, India
Sathish Kumar A.P	PSG Institute of Advanced Studies, India
Satria Mandala	Maliki State Islamic University, Indonesia
Sattar B. Sadkhan	University of Babylon, Iraq
Seemabawa	Thapar University, India
Sergio Ilarri	University of Zaragoza, Spain
Serguei A. Mokhov	Concordia University, Canada
Seungmin Rho	Carnegie Mellon University, USA
Seyed Hossein Hosseini Nazhad Ghazani	Islamic Azad University, Tabriz-Iran
Shaikh Ullah	University of Rajshahi, Bangladesh

Shin-ichi Kuribayashi	Seikei University, Japan
Shivan Haran	Arizona State University, USA
Shrirang.Ambaji.Kulkarni	National Institute of Engineering, India
Singh M.P	National Institute of Technology Patna
Somitra Sanadhya	IIT-Delhi, India
Soodamani Ramalingam	Uuniversity of Hertfordshire, UK
Sriman Narayana Iyengar	VIT University, India
Srinivas Acharyulu P.V	NTPC Limited, India
Srinivasan B	Monash University, Australia
Subashini Parthasarathy	Avinashilingam University for Women, India
Subha	M.S. University, India
Sudip Misra	Indian Institute of Technology-Kharagpur, India
Sundaresan	Manonmaniam Sundaranar University, India
SunYoung Han	Konkuk University, South Korea
Suruliandi	Manonmaniam Sundaranar University, India
Susana Sargento	University of Aveiro, Portugal
Syed Abdul Sattar	Royal Institute of Technology and Science, India
Syed Rahman	University of Hawaii-Hilo, United States
Syed Rizvi	University of Bridgeport, United States
T Venkat Narayana Rao	Hyderabad Institution of Technology and Management, India
T. Santhanam	D.G. Vaishnav College-Chennai, India
T. Devi	Bharathiyar University-Coimbatore, India
T.K.S. Lakshmipriya	Avinashilingam University-Coimbatore, India
T.V. Geetha	Anna University-Chennai, India
Tejbanta Chingtham	Sikkim Manipal Institute of Technology, India
Thamaraiselvi	Madras Institute of Technology, India
Thambidurai	Pondicherry University, India
Thooyamani K.P	Bharath University, India
Utpal Biswas	University of Kalyani, India
V. Srividhya	Avinashilingam University-Coimbatore, India
Vasantha kalyani David	Avinashilingam University-Coimbatore, India
Velmurugan Ayyadurai	Center for Communication Systems, United Kingdom
Vijayanandh R	ICIT, India
Vikram Jain	Shri Ram Institute of Technology Jabalpur, India
Vishal Sharma	Metanoia Inc, USA
Wei Jie	University of Manchester, UK
Wided oueslati	l'institut superieur de gestion de tunis, Tunisia
Xiaofeng Liao	Chongking University, China
Yan Luo	University of Massachusetts Lowell, United States
Yannick Le Moullec	Aalborg University, Denmark

Yao-Nan Lien	National Chengchi University, Taiwan
Yazid Saman	Universiti Malaysia Terengganu, Malaysia
Yeong Deok Kim	Woosong University, South Korea
Yuh-Shyan Chen	National Taipei University, TAIWAN
Yung-Fa Huang	Chaoyang University of Technology, Taiwan
Wael M El-Medany	IT College,University Of Bahrain
Raja Kumar M	National Advanced IPv6 Center (NAv6), Universiti Sains Malaysia
Sevil Sen	Hacettepe University, Turkey
Aysel Safak	Baskent University, Turkey
Jalil Jabari Lotf	Islamic Azad University, Iran
Mohammad Saleem	DERI Ireland, Ireland
Ankit Thakkar	Nirma University, India
Gagan Singla	Aryabhatta college of engineering and technology, India
Gaurav Kumar Tak	Lovely Professional University, India
J. Indumathi	Anna University, India
K. Ganesh Reddy	NITK surathkal, India
M. Sandhya	B.S. Abdur Rahman University, India
Meenu Chawla	MANIT Bhopal, India
R K Mishra	Feeroze Gandhi Institute of Engineering and Technology, India
Rama Krishna C	NITTTR Chandigarh, India
Ramesh Babu.H.S	Acharya Institute of Technology, India
Sanjay Kumar Sharma	Oriental Institute of Science & Technology, India
Santhi Thilagam	N.I.T.K-Surathkal, India
Sarbani Roy	Jadavpur University, India
T. Meyyappan	Alagappa University, India
Seckin Tunalilar	Aselsan, Turkey

Program Organizers

Chief Patron

Omer TORLAK	Rector, KTO Karatay University, Konya, Turkey

Patron

Ali Okatan	KTO Karatay University, Konya, Turkey

President

Mehmet CELIK	KTO Karatay University, Konya, Turkey

Vice - President

Hasan Fehmi ATASAGUN	KTO Karatay University, Konya, Turkey

Committee Chairpersons

Kasym Oztoprak	KTO Karatay University, Konya, Turkey
Ali Ozturk	KTO Karatay University, Konya, Turkey
Refik Caglar Kızılırmak	KTO Karatay University, Konya, Turkey
Ayben Karasu Uysal	KTO Karatay University, Konya, Turkey
Hulusi Acıkgoz	KTO Karatay University, Konya, Turkey
Mehmet Ozbay	KTO Karatay University, Konya, Turkey
Semih Yumusak	KTO Karatay University, Konya, Turkey
Adem Yılmaz	KTO Karatay University, Konya, Turkey
Mastaneh Torkamani Azar	KTO Karatay University, Konya, Turkey

Convener

Abdul Kadir Ozcan	KTO Karatay University, Turkey

Contents

Urban Traffic Management System by Videomonitoring

José Raniery Ferreira Junior

ZUQ - Intelligent Transportation
Lourival Melo Mota Ave, 12 Building, 107 Room, Federal University of Alagoas
Tabuleiro dos Martins, Maceió, Alagoas, Brasil, 57072-970
Institute of Computing, Master's Program in Informatics
Federal University of Alagoas
jose.raniery@gmail.com
http://www.zuq.com.br

Abstract. As the big cities grow, it's more necessary the use of cameras for urban traffic monitoring. The increase in the number of vehicles on the streets makes the traffic congestion, one of the largest metropolis problems, even more often to happen. To avoid this kind of issue, this paper proposes a management system by videomonitoring for the urban traffic. And the goal is to identify the vehicles e count them in period of time using Computer Vision and Image Processing techniques.

Keywords: videomonitoring, urban traffic, computer vision.

1 Introduction

Vehicle detection through videomonitoring is a important tool for real time traffic management systems. It offers some advantages against the traditional methods, such as loop detectors. Besides vehicle count, video images can provide more informations about the traffic: speed and vehicle classification [3].

So it's expectated the decrease of the congestion in the big cities and the number of accidents in the urban roads, big issues that metropolis in the whole world have to face.

The presence of Graphics Computer in the human day-by-day is increasing, and even more Computer Vision and Image Processing. The computers and cameras prices are the lowest ever seen and it gets more viable to equip the roads with an artificial vision system that is able to detect movements, follow vehicle track and extract informations, such as speed, dimensions and traffic density [1].

Some useful algorithms for vehicle detection in the literature were: background classification learning [3], image segmentation [1], application of mathematical morphology operations, such eroding and dilation [4], others [5] [2].

The goal of this paper is to develop a system that aids the traffic management of main urban roads and aim to decrease the number of vehicles. This way, the expectation is the congestions end by means of vehicle detection and count.

D. Nagamalai et al. (Eds.): *Adv. in Comput. Sci., Eng. & Inf. Technol.*, AISC 225, pp. 1–9.
DOI: 10.1007/978-3-319-00951-3_1

As secondary goal, it expects that the algorithm has low processing cost, using opensource tools.

2 Methods

All video samples used in this system had speed rate of 30 frames per seconds and dimensions 640x480 coloured pixels.

The system was developed in Java 1.7 programming language, along with OpenCV (Open Source Computer Vision) version 2.4.0 graphic library with a wrapper to Java called JavaCV. OpenCV is an opensource library useful in several areas and techniques, as Computer Vision, Image Processing and Segmentation, Machine Learning, and others [6].

The system runs some steps, listed below. Bradski [7] affirms that the Warping operation is a geometric transformation based on image not uniform resizing. The Figure 1 shows a Warping operation example. In the system algorithm case, four points of the image are selected to form a trapezoid. This trapezoid corresponds to the frame's region of interest (ROI). The ROI is the image area where vehicle recognition happens. In the system case, only part of the street is the region of interest (Figure 2(a)). Vehicles that are far away are harder to identify, because of the proximity of itself and other vehicle. The areas that correspond to the sidewalk and the begin of the street are not of interest for processing. After the trapezoid's definition, the Warping operation is done and the image becomes the whole area corresponding to the trapezoid (Figure 2(b)).

1. Warping of the region of interest of the frame
2. Image conversion in grayscale
3. Background removal
4. Image binarization
5. Eroding and Dilation application
6. Contour areas detection
7. Count of the areas

Following, the image conversion in grayscale is performed, a necessary step to achieve better results in posterior steps (binarization e.g.). After that, the image background is removed, to identify only the moving objects (vehicles). This operation is performed by the OpenCV's function *cvAbsDiff*. This function performs the subtraction of two images (original e background), resulting the foreground. After that, the binarization operation is performed, so the image is segmented with 128 threshold, resulting in a black and white image (Figure 3(a)).

The mathematical morphology operations, eroding e dilation, are executed in sequence to eliminate isolated "spots" that do not make part of a vehicle and to group the "spots", respectively, that make part of the same vehicle, but somehow are separated (Figure 3(b)).

The localization of white "spots" contours is the next step of the algorithm. It is done by the OpenCV's function *cvFindContours*. As it was locating the "spots" frame by frame, it was able to perform the vehicle count by following:

(a) Original Image (b) Image after Warping operation

Fig. 1. Warping Operation

(a) ROI marked by red trapezoid (b) Image after Warping operation of the ROI

Fig. 2. Warping of the region of interest of the frame

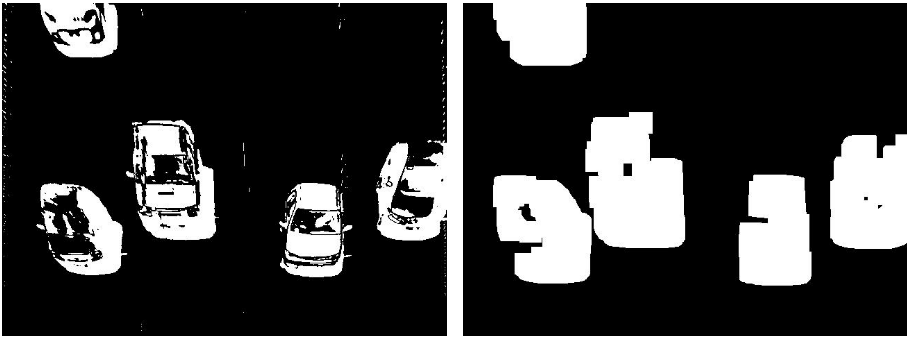
when a new vehicle “is born” in the frame, a rectangle is designed and the vehicle is put within it. After that, the vehicle is followed until the last frame which the “spot” appears. By the moment the rectangle is designed, it is calculated the rectangle centroid or its central point (see Formula 1). When a centroid crosses the counting line, a new vehicle is registered and counted.

$$xCentroid = \frac{w}{2} + x;$$

$$yCentroid = \frac{h}{2} + y;$$

Formula 1. Point (xCentroid, yCentroid) of the rectangle’s centroid.

The centroid is a point (xCentroide, yCentroide), and this point is calculated by the formula above, where (x,y) is the far left and high point of the “spots” rectangle, w is the rectangle width and h is the height.



(a) Binary image of moving vehicles (b) Binary image after the morphological operations

Fig. 3. Result of morphological operations in a binary image

After the "birth" of a "spot", it is attributed an id to it. In the next frame, it is checked if this centroid is located in the rectangle interior of some "spot" in the previous frame. If it is, that's because it's the same "spot", than it is the same id. If it's not, than it's a new one, than it's attributed a new id for it.

2.1 Shadow Removal

Shadows can cause some problems in background subtraction and vehicle detection, as you can see in Section 3 (Results and Discussion) of this paper. The detection of cast shadows as foreground objects is very common, producing undesirable consequences: shadows can connect different people walking in a group, generating a single object as output of background subtraction [8], but there's a technique in literature that can eliminate them of the grayscale image. Jacques [8] affirms that the normalized cross-correlation (NCC) can be useful to detect shadow pixel candidates.

As Jacques [8] described, *B(i,j)* is the background image and *I(i,j)* is an image of the video sequence. For each pixel *(i,j)* belonging to the foreground, consider a $(2N + 1)$ X $(2N + 1)$ template T_{ij} such that $T_{ij}(n, m) = I(i + n, j + m)$, for $-N \leq n \leq N$, $-N \leq m \leq N$ (T_{ij} corresponds to a neighborhood of pixel *(i,j)*). Then, the NCC between template T_{ij} and image B at pixel *(i,j)* is given by Formula 2:

$$NCC(i,j) = \frac{ER(i,j)}{E_B(i,j)E_{Tij}},$$

where

$$ER(i,j) = \sum_{n=-N}^{N} \sum_{m=-N}^{N} B(i+n, j+m)T_{ij}(n,m),$$

$$E_B(i,j) = \sqrt{\sum_{n=-N}^{N} \sum_{m=-N}^{N} B(i+n, j+m)^2}, and$$

$$E_{Tij} = \sqrt{\sum_{n=-N}^{N} \sum_{m=-N}^{N} T_{ij}(n,m)^2}.$$

Formula 2. Normalized cross-correlation in a point (i, j).

A pixel *(i,j)* is pre-considered shadow if:

$$NCC(i,j) \geq L_{ncc}$$

and

$$E_{Tij} < E_B(i,j)$$

Formula 3. Condition for a point (i, j) to be pre-considered shadow.

where L_{ncc} is a fixed threshold. In this system, $N = 4$ and $L_{ncc} = 0.999$. Then, pixels that are false positive are eliminated. This stage consists of verifying if the ratio I(i, j)/B(i, j) in a neighborhood around each shadow pixel candidate is approximately constant, by computing the standard deviation of I(i, j)/B(i, j) within this neighborhood. More specifically, we consider a region R with $(2M+1)(2M+1)$ pixels (it was used $M = 1$ in all experiments) centered at each shadow pixel candidate *(i, j)*, and classify it as a shadow pixel if:

$$std_R(\frac{I(i,j)}{B(i,j)}) < L_{std}$$

and

$$L_{low} \leq (\frac{I(i,j)}{B(i,j)}) < 1$$

Formula 4. Condition to a point (i, j) to be a shadow pixel.

where $\mathrm{std}_R(\frac{I(i,j)}{B(i,j)})$ is the standard deviation of quantities I(i,j)/B(i,j) over the region R, and L_{std}, L_{low} are thresholds. In this system, it was used $L_{std} = 0.05$ and $L_{low} = 0.3$. Then it was applied morphological operations to enhance the results. As experiment, it was used a frame with three "spots", and after the shadow removal algorithm has been applied, the "spots" decreased the area in 25%, 18% and 19%, as it can be seen in Figure 4.

Despite this results, the shadow removal algorithm was not used in the main one because it demands much processing time (it takes some seconds to process a single frame), and one of the goals of the system is to have a low processing cost. So the algorithm will be enhanced, so in the future it can be used in the main algorithm.

(a) "Spots" before shadow removal algorithm

(b) "Spots" after shadow removal algorithm

Fig. 4. Result of shadow removal

3 Results and Discussion

Figure 5 shows a frame completely processed, so it was performed all the described operations before, needed to recognize vehicles. In the Figure 5, six vehicles are shown, all identified (with rectangle and centroid), but only two were counted, both inferior, as only their centroids were below the counting line (green colour). In the next frame, if these same vehicles are still in the frame, they are not counted again. If other vehicle's centroid crosses the line, it will be counted one more vehicle.

It was performed tests with three video sample. The results are in the Table 1. It was computed the processing times of vehicle detection for each frame. The PMAI column of the table shows the biggest time taken by a frame to detect all vehicles. PMEN is the lowest time and PMED, the mean time of all frames of the sample. The NVT column of the table is the total number of vehicles the is shown in the video sample, independent if the vehicle crossed the counting line or not. NVM is the total number of vehicles that crossed the counting line, under manual count. The number of vehicles that crossed the green line that were counted by the algorithm can be seen by the NVA column.

Table 1. Sample results

Sample	Duration	Number of frames	PMAI	PMEN	PMED	NVT	NVM	NVA
1	15 s	450	41 ms	5 ms	9 ms	23	21	21
2	30 s	900	32 ms	5 ms	5 ms	27	22	23
3	60 s	1800	49 ms	5 ms	7 ms	74	69	63

The sample 1 showed a processing mean bigger than all samples, but all the vehicles that crossed the counting line were counted. Just two vehicles stayed

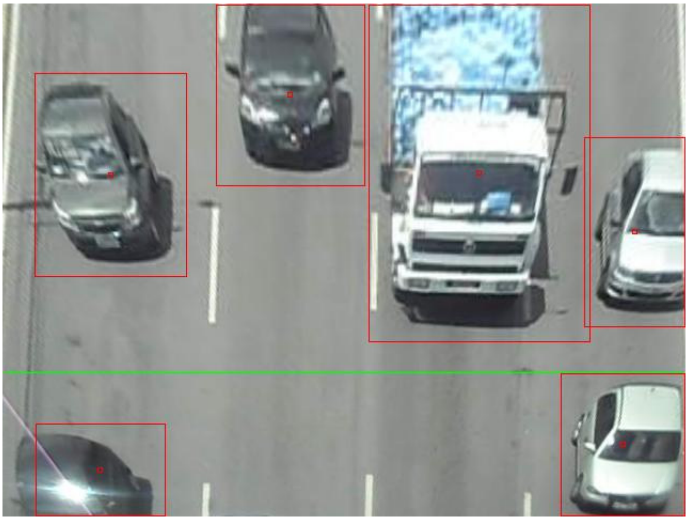


Fig. 5. Recognition of vehicles in a frame

above the counting line, so that they were not registered. In sample 2, video duration was bigger, but the processing time for the recognition of the vehicles was lower, considering that its mean was 5 milliseconds, against 9 milliseconds for the first sample. It was counted 27 vehicles in total, but only 22 crossed the counting line.

The sample 3 had bigger video duration, and still a processing time mean lower than the sample 1. That means that sample 1 reached a bigger number of "spots" by frame than all others samples, but by the time the time duration was lower than the others, the number of vehicles also was lower.

Still about the sample 3, it was counted 74 vehicles manually, but only 69 could be counted because they were below the counting line. The algorithm counted 63 vehicles below the line.

Some error cases in the algorithm were found, such:

- "Spots" inside others are not considered;
- "Spots" with small areas, that are not vehicles, are not counted;
- "Spots" with large width (likely two cars side by side that were not separated), the area was divided in two, so it will count two vehicles;
- Vehicles that "are born" with two "spots" and than become one, is counted only once (the algorithm counts two times because of the two initial "spots");
- Vehicles that "are born" with one spot and then becomes two, are counted only once.

Nevertheless, in this last case, there is a problem: when two vehicles are too close to each other in the "birth", only one "spot" is considered because of the shadow of both. After a while, much times the "spot" of two vehicles splits. So, by the previous rule, it will not count the second car as a new one, considering that only one "spot" in the beginning.

3.1 Tests

To test the accuracy of the algorithm, it was created a comparison mechanism of the manual and the algorithm count. Given a short interval of time, five seconds e.g., it is counted the number of vehicles, both manually and by the algorithm, that crosses the counting line. Then these counts are compared and it is obtained some indicators, like: difference between the manual and the algorithm counts, the total number of errors of the algorithm, the interval of time that has the larger number of errors, absolut mean, relative mean, variance and standard deviation of the errors, among others.

Some of these results and indicators are listed below in Table 2:

Table 2. Tests results

Sample	Duration	Number of Frames	NIE	TNE	AME	RME	Variance	SD
1	60 s	1800	7	8	1.14	0.51	0.12	0.35
2	160 s	4800	27	78	2.89	0.64	4.54	2.13

NIE column represents the number of intervals that have errors in the whole video sample. TNE is total number of errors in all intervals. AME and RME are the absolute and the relative mean of errors that a interval has, respectively. SD is the standard deviation of all errors in the video sample.

4 Conclusion

This paper presents a systems with low processing cost algorithm and success rate of 90% like showed in the three video samples in Table 1. Also it has good perspectives for the future to enhance the algorithm, considering that shadow removal technique achieved good initial results.

To obtain a bigger success rate, the ideal view would be a camera in the vertical position or 90 degrees angle towards the street and not diagonal, like presented in the video samples. So that, vehicles very close to each other would not be connected in a non appropriated way.

The disadvantages of the algorithms are: the need to obtain a background image manually and the difficulty to separate vehicles that are very close to each other. For future works, it is proposed an algorithm to eliminate the image background automatically and to integrate the shadow algorithm to the main one.

References

1. Haupt, A.G.: Detecção de Movimento, Acompanhamento e Extração de Informações de Objetos Móveis. Universidade Federal do Rio Grande do Sul (2004)
2. Gupte, S., Masoud, O., Martin, R.F.K., Papanikolopoulos, N.P.: Detection and Classification of Vehicles. IEEE Transactions on Intelligent Transportation Systems (2002)
3. Tan, X., Li, J., Liu, C.: A video-based real-time vehicle detection method by classified background learning. World Transactions on Engineering and Technology Education (2002)
4. Purnama, I.K.E., Zaini, A., Putra, B.N., Hariadi, M.: Real Time Vehicle Counter System for Intelligent Transportation System. In: International Conference on Instrumentation, Communications, Information Technology, and Biomedical Engineering (2009)
5. Daigavane, P.M., Bajaj, P.R.: Real Time Vehicle Detection and Counting Method for Unsupervised Traffic Video on Highways. International Journal of Computer Science and Network Security (2010)
6. OpenCV Official Homepage, `http://opencv.willowgarage.com/wiki/` (last access: July 26, 2012)
7. Bradski, G., Kaehler, A.: Learning OpenCV - Computer Vision with the OpenCV Library. O'Reilly Publisher (2008)
8. Jacques Jr., J.C.S., Jung, C.R.: Background Subtraction and Shadow Detection in Grayscale Video Sequences. In: Proceedings of the XVIII Brazilian Symposium on Computer Graphics and Image Processing (2005)

Symbolic Classification of Traffic Video Shots

Elham Dallalzadeh[1], D.S. Guru[2], and B.S. Harish[3]

[1] Department of Computer, Marvdasht Branch, Islamic Azad University, Marvdasht, Iran
elhamdallalzadeh@gmail.com
[2] Department of Studies in Computer Science, University of Mysore, Manasagangothri, Mysore - 570 006, Karnataka, India
dsg@compsci.uni-mysore.ac.in
[3] Department of Information Science and Engineering, S J College of Engineering, Mysore, Karnataka, India
bharish@ymail.com

Abstract. In this paper, we propose a symbolic approach for classification of traffic video shots into light, medium, and heavy classes based on their content (congestion). We propose to represent a traffic video shot by an interval valued features. Unlike the conventional methods, the interval valued feature representation is able to preserve the variations existing among the extracted features of a traffic video shot. Based on the proposed symbolic representation, we present a symbolic method of classifying traffic video shots. The symbolic classification method makes use of a symbolic similarity measure for classification. An experimentation is carried out on a benchmark traffic video database. Experimental results reveal the efficacy of the proposed symbolic classification model. Moreover, it achieves classification within negligible time as it is based on a simple matching scheme.

Keywords: Traffic congestion, classification of traffic video shots, symbolic representation, interval valued features, symbolic similarity measure, symbolic classifier.

1 Introduction

Traffic congestion is a serious issue in many urban streets and highways. In order to reduce the traffic congestion, traditional solutions have been employed that are based on increasing the supply of roads. However, such solutions are costly. Furthermore, it is not always possible to employ because of the lack of suitable lands. Recently, interest has been grown up in optimizing the throughput of the existing roads using vision-based traffic monitoring systems.

Vision-based traffic video monitoring systems help us in gathering statistical data on traffic activity through monitoring the density of vehicles. Furthermore, it assists taking intelligent decisions and actions in any abnormal conditions by analyzing the traffic information. Hence, an intelligent automated monitoring system can detect

D. Nagamalai et al. (Eds.): *Adv. in Comput. Sci., Eng. & Inf. Technol.*, AISC 225, pp. 11–22.
DOI: 10.1007/978-3-319-00951-3_2

traffic jams and divert vehicles from congested roads to less crowded ones. On the other hand, it is a key task to collect information on traffic flows in real-time.

Most of the information required for classification of traffic video shots can be based on motion that each traffic video contains. Thus, a holistic representation can be used to capture the variability of the motion without the need for segmenting or tracking individual components.

Yu et al., [2] presented a novel algorithm to directly extract highway traffic information such as average vehicle speed and density from MPEG compressed Skycam video. The MPEG motion vector field is filtered to remove vectors that are not consistent with vehicle motion. The traffic flow is then estimated by averaging the remaining motion vectors.

Porikli and Li [3] proposed an unsupervised, low-latency traffic congestion estimation algorithm that operates on the MPEG video data. Congestion features are directly extracted from the compressed domain. Gaussian Mixture Hidden Markov Models (GM-HMM) is employed to detect traffic congestion. Traffic patterns are detected as five traffic patterns (empty, open flow, mild congestion, heavy congestion, and stopped).

A framework to collect traffic flow information from urban traffic scenes is proposed by Lee and Bovik [1]. The traffic flow is presented by defining traffic regions. Basic statistics of traffic flow vectors inside the traffic regions are then computed.However, extracting reliable measurements of flow is difficult in traffic scenarios due to environmental conditions. Moreover, the extracted measurements are subjected to noise.

Chan and Vasconcelos [4] proposed an approach to model the entire motion field as a dynamic texture. It is an auto-regressive stochastic process with both spatial and temporal components. The auto-regressive stochastic process encodes the appearance and the underlying motion separately into two probability distributions. Distances among dynamic textures are computed using information theoretical measures of divergence (Kullback-Leibler divergence) between the associated probability distributions or through geometry measures based on their observable space (Martin distance). With these distance measures, the traffic congestion is classified using a Nearest Neighbor (NN) classifier or by the use of a Support Vector Machine (SVM) classifier with the Kullback-Leibler kernel. However, the proposed model is computationally complex. It cannot be well-adopted for on-line and real-time estimation of traffic congestion.

Derpanis and Wildes [5] described a system based on 3D spatio-temporal oriented energy model of dynamic texture. Traffic patterns are classified in terms of their dynamics measures aggregated over regions of image space-time. For such purposes, local spatio-temporal orientation is derived. Each traffic scene is then associated with a distribution (histogram) of measurements. Classification is performed based on matching such distributions (or histograms) of space-time orientation structure. However, heavy traffic in which the traffic stopped completely cannot be distinguished from light traffic with almost an empty (stationary) roadway.

The above mentioned issues motivate us to propose a simple yet efficient model to classify traffic video shots based on their content (congestion). We outline a novel method of representing and classifying traffic video shots using symbolic data analysis concepts.

The remaining part of the paper is organized as follows. In section 2, we discuss the proposed model for symbolic representation and classification of traffic video shots. The details of an experimentation conducted to demonstrate our proposed model on UCSD benchmark traffic video database are given in section 3. The paper is concluded in section 4.

2 Proposed Model

In this section, we outline a novel method of representation and classification of traffic video shots based on their content (congestion) using symbolic data concepts. Initially, each traffic video shot is filtered using Gabor filter to segment the texture content of its frame sequence. We make use of motion, appearance, and texture features as the conventional features to construct feature matrices for the frame sequence of a traffic video shot.

However, the extracted feature matrices contain lots of intra-class variations. On the other hand, there exists low inter-class variations among such feature matrices extracted from various traffic video shots of different classes. To capture the intra-class variations as well as representing traffic video shots of different classes effectively, an interval valued feature vector representation is formulated to represent each traffic video shot. Traffic video shots are then classified based on symbolic similarity measure.

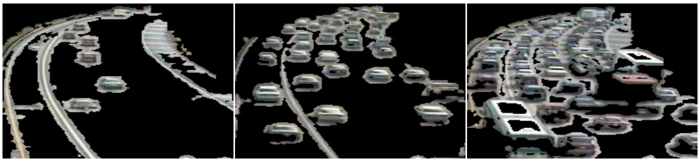
2.1 Pre-processing

Given a frame sequence of a traffic video shot, the pre-processing step is done to segment the content of the frames by their texture. A Gabor filter [8] with different frequencies along with predefined orientations is applied on the RGB channels of each of the frames of a traffic video shot. Each of these filters gives an image as a two-dimensional array of the same size as the input frame. The magnitude of each of the Gabor filtered images is taken and added to each other to get a final filtered image for a given frame. The obtained filtered image is then mapped on the given frame to segment its texture content. The margin of the segmented frame is then cropped to have only the content of the frame.

The above mentioned pre-processing step is employed to segment the content of the frame sequences of traffic video shots. The segmented texture content of the sample frames of three different traffic video shots are shown in Fig. 1.

(a)

(b)



(c)

Fig. 1. Frame segmentation of traffic video shots. (a) Sample frames of light, medium, and heavy traffic, (b) Final Gabor filtered images of the frames, (c) Segmented texture content of the frames after mapping the filtered images on the sample frames and cropped the margin of the segmented frames.

2.2 Symbolic Feature Representation

To gather the statistical data on a traffic activity, the content of traffic video shots can be represented by motion. However, the motion of traffic video shots has a limited ability to distinguish between heavy traffic (almoststopped traffic) and lighttraffic(almost empty/stationary roadway). The estimated motion of such traffics is nearly the same. Moreover, the estimated motion of heavy traffic with slight speed cannot be well classified from the medium traffic at reduced speed.

Therefore, we propose to represent the content of a traffic video shot by appearance and texture in addition to motion. Hence, motion, appearance, and texture features of the content of traffic video shots are extracted.

Features Based on Motion

Motion estimation is the process of determining motion vectors that describe the transformation from one 2D image to another. We propose to estimate the motion of a traffic video shot using a global pixel-based method by considering the whole frame as one block. Sum of absolute differences (SAD) is considered as an evaluation metric to compute the motion of a frame with respect to its previous one.

Let $\{F_1, F_2, F_3, \ldots, F_R\}$ be a set of R frames of a traffic video shot. Considering the k^{th} frame, say F_k, the motion of a frame F_k with respect to the previous frame F_{k-1} is calculated as,

$$MF_k = \sum_{x=1}^{u} \sum_{y=1}^{v} \left| F_k(x,y) - F_{k-1}(x,y) \right| \tag{1}$$

where, $F_k(x,y)$ and $F_{k-1}(x,y)$ are the intensity values of the pixels at (x,y) of the frames F_k and F_{k-1} respectively. Also, u and v define the size of the frames.

Therefore, a column vector called motion feature vectorsay MFof size R-1, considering the adjacent frames, is obtained.

In order to have a compact representation of the obtained motion feature vector in addition to capturing the variations of the motion feature values, we propose to represent the motion feature vector in the form of an interval valued feature vector. Though, it shall be noted that motion is a dominant feature for content representation of a traffic video shot.

Hence, we propose to symbolically represent the motion feature vector by preserving the variations of the features as well as conserving the domain of the features. So, we end up having an interval valued feature vector of dimension 2 representing motion of a traffic video shot.

An interval valued motion feature preserving the variations among the motion feature values is represented as,

$$\text{Motion\_Interval1} = \left[MF^{-}, MF^{+} \right] \tag{2}$$

where,

$$MF^{-} = \mu(MF) - \tau(MF) \quad \text{and} \quad MF^{+} = \mu(MF) + \tau(MF) \tag{3}$$

where, $\mu(MF)$ is the mean of the motion feature values, $\sigma(MF)$is the standard deviation of the motion feature values, and $\tau(MF)$is the function of $\sigma(MF)$given by $\tau(MF) = \alpha \times \tau(MF)$ for some scalar value αthat is set empirically.

Similarly, another interval valued motion feature capable of conserving the domain of the motion feature values is obtained as,

$$\text{Motion\_Interval2} = \left[MF^{-}, MF^{+} \right] \tag{4}$$

where,

$$MF^{-} = \min(MF) \quad \text{and} \quad MF^{+} = \max(MF) \tag{5}$$

where, min(MF)andmax(MF)are the minimum and maximum of the motion feature values.

Hence, the motion feature vector is symbolically represented by two interval valued features named as Motion_Interval1 and Motion_Interval2.

Features Based on Appearance

To represent the distribution of the data values in a traffic video shot, a histogram of each of the frame is extracted. Considering the k^{th} frameF_k, the RGB histograms of the frame F_k are represented by $\mathcal{H}$ number of gray levels, in here 256 gray levels, as the number of bins. The represented color channel histograms are assimilated and represented as a single appearance distribution vector for the frame F_k by considering the maximum frequency of the color channels.

A matrix called appearance feature matrix, say AFof size R×$\mathcal{H}$ (here R×256), is obtained. Each row is an appearance distribution vector representing its respective frame.

Instead of keeping this huge feature matrix, we recommend capturing the variation of the entire matrix in the form of an interval valued. Thus, we end up having an interval valued feature representing the appearance of a traffic video shotgiven by,

$$\text{Appearance\_Interval} = \left[AF^{-}, AF^{+} \right] \tag{6}$$

where, AF^{-} and AF^{+} are the limits of the interval computed as in Equation (3).

Hence, the appearance feature matrix is symbolically represented by an interval valued feature called Appearance_Interval.

Features Based on Texture

A statistical method of examining textures that consider the spatial relationship of pixels is the gray level co-occurrence matrix (GLCM), also known as the gray level spatial dependence matrix. The GLCM matrix can reveal certain properties about the spatial distribution of the gray levels in the texture image. The GLCM function calculates how often a pair of pixels with specific intensity (gray level) values and in a specified spatial relationship occur in an image [9].

We use a GLCM functionwhich creates multiple GLCMs for a single input frame. Thus, an array of offsets for the GLCM function is identified. These offsets define pixel relationships of varying directions and distances. After generating the GLCMs, four statistical textural features [9] such as contrast, correlation, energy, and homogeneity are derived.

Considering the k^{th} frameF_k, the four GLCMs of the frame F_k are obtained by defining an array of offsets specifying four directions (0°,45°, 90°, 135°) with distance 1. For each of the obtained GLCMs, the statistical textural features such as contrast, correlation, energy, and homogeneity are calculated. In addition to these statistical textural features, the entropy of the frame F_k is also estimated as an added feature, thereby totally 17 features.

In general, $\mathcal{T}$number of statistical texture features are extracted for the frame F_k. So, a texture vector of size $\mathcal{T}$ is created (here $\mathcal{T}$=17). Furthermore, it has to be noticed that all the 17 statistical texture features are of different nature.

A matrix called texture feature matrix, say TFof size R×$\mathcal{T}$ (here R×17), is obtained.

To compress the representation of this matrix and to capture the variations of the extracted features, we recommend capturing the variations in each column in the form of an interval valued. Thus, we end up having 17 interval valued features given by,

$$\text{Texture\_Interval\_Vecor} = \left[TF_1, TF_2, ..., TF_{17} \right] \quad (7)$$

where, each TF_i, i =1, 2, …, 17, is an interval valued feature computed as in Equation (3) for the respective column of TF.

2.3 Traffic Video Shots Representation and Classification

We propose to represent the traffic video shots in the knowledgebase by their reference feature vectors. The reference feature vectors are represented by assimilating the obtained interval valued features discussed in subsection 2.2. The reference feature vectors are used in classification.

Given a test sample traffic video shot, its similarity values with respect to all the reference feature vectors stored in the knowledgebase are computed. The test sample traffic video shot is then labeled by the class label of a traffic video shot with a maximum similarity.

Representation

Let there be O number of traffic video shots to be represented in the knowledgebase. A traffic video shot, say T_j (j =1, 2, 3, …, O), is represented by a reference feature vector, say Ref_jgiven by,

$$Ref_j = \left(\text{Motion\_Interval 1, Motion\_Interval 2, Appearance\_Interval, Texture\_Interval\_Vector} \right) \quad (8)$$

Hence, a vector of interval valued features of dimension 20 is formed. Similarly, each traffic video shot is represented by its reference feature vector. Thus, O number of reference feature vectors are created and stored in the knowledgebase.

Classification

We propose to compute the similarity of a given test sample traffic video shot, say T_t, with respect to all the reference feature vectors stored in the knowledgebase by the use of symbolic similarity measure [6]. The similarity is estimated between the feature vector of a test sample and the reference feature vectors stored in the knowledgebase.

Thus, it is proposed to represent a test sample traffic video shot T_t by a crisp feature vector, say VF.

$$VF = \left(\text{Motion\_mean, Motion\_median, Appearance\_mean, Terxture\_mean\_Vector} \right) \quad (9)$$