1-1-2014

Computer vision based traffic monitoring system for multi-track freeways

Zubair Iftikhar
University of Wollongong, zi770@uowmail.edu.au

Prashan Premaratne
University of Wollongong, prashan@uow.edu.au

Peter James Vial
University of Wollongong, peterv@uow.edu.au

Computer vision based traffic monitoring system for multi-track freeways

Abstract

Nowadays, development is synonymous with construction of infrastructure. Such road infrastructure needs constant attention in terms of traffic monitoring as even a single disaster on a major artery will disrupt the way of life. Humans cannot be expected to monitor these massive infrastructures over 24/7 and computer vision is increasingly being used to develop automated strategies to notify the human observers of any impending slowdowns and traffic bottlenecks. However, due to extreme costs associated with the current state of the art computer vision based networked monitoring systems, innovative computer vision based systems can be developed which are standalone and efficient in analyzing the traffic flow and tracking vehicles for speed detection. In this article, a traffic monitoring system is suggested that counts vehicles and tracks their speeds in realtime for multi-track freeways in Australia. Proposed algorithm uses Gaussian mixture model for detection of foreground and is capable of tracking the vehicle trajectory and extracts the useful traffic information for vehicle counting. This stationary surveillance system uses a fixed position overhead camera to monitor traffic.

Keywords

system, monitoring, track, traffic, multi, vision, computer, freeways

Disciplines

Engineering | Science and Technology Studies

Publication Details

Z. Iftikhar, P. Premaratne & P. Vial, "Computer vision based traffic monitoring system for multi-track freeways," Lecture Notes in Computer Science, vol. 8589, pp. 339-349, 2014.

Computer Vision Based Traffic Monitoring System for Multi-Track Freeways

Zubair Iftikhar, Prashan Dissanayake and Peter Vial

School of Electrical Computer and Telecommunications Engineering,
University of Wollongong, North Wollongong, NSW, Australia
zi770@uowmail.edu.au

Abstract. Nowadays, development is synonymous with construction of infrastructure. Such road infrastructure needs constant attention in terms of traffic monitoring as even a single disaster on a major artery will disrupt the way of life. Humans cannot be expected to monitor these massive infrastructures over 24/7 and computer vision is increasingly being used to develop automated strategies to notify the human observers of any impending slowdowns and traffic bottlenecks. However, due to extreme costs associated with the current state of the art computer vision based networked monitoring systems, innovative computer vision based systems can be developed which are standalone and efficient in analyzing the traffic flow and tracking vehicles for speed detection. In this article, a traffic monitoring system is suggested that counts vehicles and tracks their speeds in realtime for multi-track freeways in Australia. Proposed algorithm uses Gaussian mixture model for detection of foreground and is capable of tracking the vehicle trajectory and extracts the useful traffic information for vehicle counting. This stationary surveillance system uses a fixed position overhead camera to monitor traffic.

Keywords: computer vision. surveillance system. Gaussian mixture model. vehicle trajectory

1 Introduction

Computer vision is effectively used in manufacturing industry for assembling electronics control in vehicles by robots. Quality control in multi-Billion dollar electronics industry is maintained by computer vision where involvement of the human eye is unheard of during the past decade. Large infrastructure projects are too vast to monitor by humans along. More and more computer vision based systems are developed for security and maintenance. Computer vision has been successfully used in the UK for monitoring traffic to avoid traffic bottlenecks. However, such systems cost tax payers millions of dollars every year and are far from economical in the current economic slowdown of the world. So, it is very desirable to develop new surveillance systems which are stand along and inexpensive that making use of the latest hardware developments even for smaller freeways. In a traffic monitoring system, vehicles can be detected and tracked for their speed for a short period of time

using a live video stream. This can be used for counting the traffic density of any track or section of the freeway and could potentially identify the traffic jams due to certain kind of breakdown or accidents. The propose can detect the event before the authorities are notified of any event. Major applications of such systems include intelligent traffic monitoring, counting vehicle on road, traffic rule violation detection, classification of E-TAG (Electronic tag) system. In recent years, there have been few instances of intelligent vehicles with autonomous driving. However, there are many challenges for such a system in practice. Major challenges in vehicle detection algorithm are weather conditions, poor road illumination, occlusions from other vehicles, view point orientation and the challenges posed by variety of vehicles with trailers. A lot of work has been attempted to handle some of these issues.

Vehicle detection can be classified as camera based systems or optical sensor based detection and tracking [1]. Computer vision has been successfully implemented in many vision-related scenarios with increasing reliability [2-11]. Camera quality and its sensing system have been improved a lot in recent times and same level of improvement can be seen in computational power of computer system. These factors are enabling researchers to enhance vehicle detection algorithm's accuracy. Number of sensors and quality of sensing device are key factors in computer vision application. Major goal of many current research activities regarding vehicle monitoring is geared towards improving accuracy with reduction in complexity to handle different weather conditions and occlusion problems.

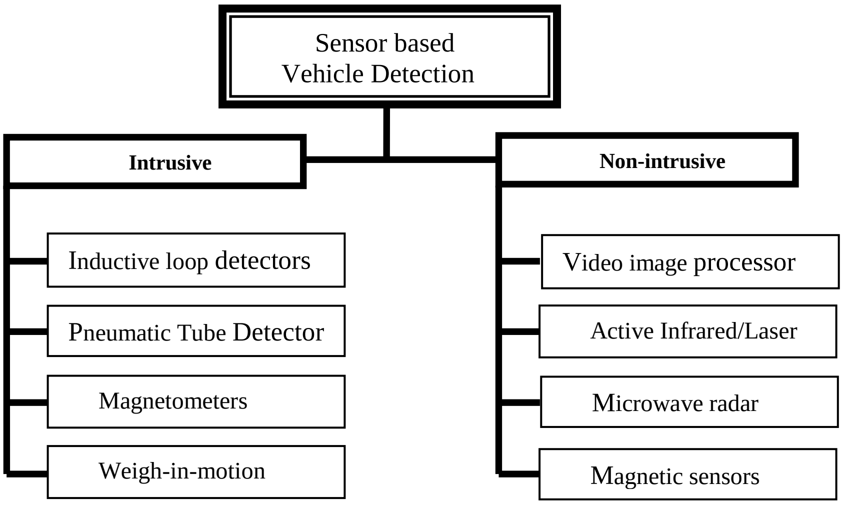


Fig. 1. Categorization of Sensor based Vehicle Detection Methods

1.1 Sensor Based Vehicle Detection

Sensor based vehicle detection system is a non-computer vision based system. It is based on the physical interaction or presence of vehicles. It is categorized in two

major types as shown in Fig. 1. Intrusive techniques are based on the physical interaction of vehicles with sensors. In these approaches, embedded sensor detects the existence of vehicle through weight or metal body of vehicle. Second major type is non-Intrusive; it is based on the vehicle presence.

1.2 Vision Based Vehicle Detection

Out of all the existing technologies of traffic surveillance, computer vision-based systems are one of the latest and most considerable. Numerous research work have been carried out in computer vision-based Intelligent Transportation Systems **(**ITS) due to its various features as compare to others such as easy installation and maintenance, quick response, simple operation and maintenance, high flexibility, low cost, and their ability to monitor and analyze of wide areas [2-11]. Major issues in vision based system are occlusion and illumination. The common approach in vision based algorithms is to apply preprocessing such as filtering to remove sensor noise followed by detecting the foreground. Then, segmentation of vehicle position is attempted through connected component analysis. Finally, detected vehicle is tracked. In literature, vision based multi vehicles detection algorithm are classified as Background Detection [12], Color Cluster method [13], Graph Axis method [14], Optical flow [15] and Space Vector Differencing.

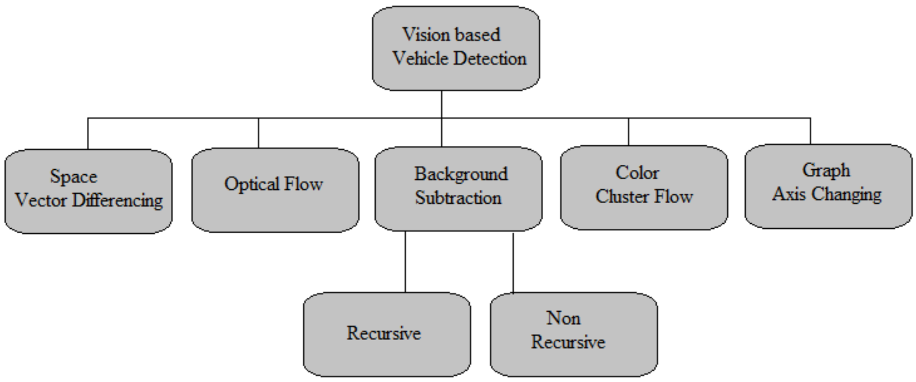


Fig. 2. Categorization of Vision based Vehicle Detection Methods

Background modeling approaches [16][17] are very popular and widely used because these techniques supply the most complete feature data for accurate detection of moving objects. Background modeling methods develop the background model by observing the scene for a particular time period. After background modeling, subtraction of current frame from the background frame is performed for detecting the moving objects. Accurate building of background model is a key challenge especially in dynamic outdoor environment. In addition, it is necessary to adapt to new conditions of weather, illumination and shadows and update the model. Background modeling approach has a very straightforward mechanism. This approach assumes that background scene will not change with sequence of frames.

Firstly, difference D_k from the current frame f_k and the background model frame b_k is determined by using equation (1) [18].

$$D_k(x,y) = | f_k(x,y) - b_k(x,y) | \tag{1}$$

Then threshold the difference D_k according to equation (2) [18]:

$$\mathrm{R}_k(x,y) = \begin{Bmatrix} 1, \mathrm{D}_k(x,y) > \mathrm{T} \\ 0, \text{otherwise} \end{Bmatrix} \tag{2}$$

If R_k has value '1' then this pixel is foreground pixel. Selection of threshold value *T* has direct impact on the quality of background modeling. Too high *T* affects as broken occurrence on region of targeted moving object. Whereas, too low value of *T* introduces a lot of noise. Generally, threshold is chosen through gray histogram by finding double peaks or more and then selecting the bottom value between two peak values [18]. Foreground detection is an essential step in vision based detection. In literature, recursive and non-recursive approaches are presented to detect foreground.

Non recursive approaches are further classified in frame differencing [19], median filter [20], and linear predictive filtering [21]. Frame differencing [19] technique uses previous frame of time instance *t-1* as the reference for current frame at time *t*. This method might not able to determine the interior pixels of colored object movement. In median filter approach, the estimated background is defined as the median value for each pixel position of every frame. Assuming that pixel remains in background about in half of the frame. Median filtering [20] uses median for color images. Estimation of background is performed using a linear prediction filter [21] to the pixels in the buffer. Filter coefficients are evaluated at each frame time on the basis of the covariance of the sample, which makes it difficult to implement in real time. Recursive approaches are classified in Approximated Median [20], Kalman filter [22] and Mixture of Gaussian [23]. In Approximated Median [20], estimation of the center is increased by one as a result of input pixel value is greater than the estimated value, and vice versa. Estimation converges to a point where half of the input pixels are higher than the others. Kalman filter [22] is a recursive technique for tracking of linear dynamical systems with Gaussian noise. Kalman filter estimates the global illumination change, and noise variance. Also, it performs the management of structures pixel. In mixture of Gaussian approach, pixel values that are not belonged to background are assumed as foreground. This is a famous technique for segmentation ofmoving regions at real-time. Gaussian model are updated using *K*-means approximation method. Each Gaussian distribution is appointed to represent the background or a moving object in the model of adaptive mixture. Each pixel is then evaluated and classified.

Color cluster method [13] combines the information of different sources, such as structure and color. The information structure is the census transforms and color information which is obtained from color histograms. Color image is divided into method clusters. Objects of interest are grouped based on color contrast so object shape is reflected by the respective color shape. To this end, it must have the prior knowledge and data of individual objects.

In graph axis method [14], pixels of an image are modified according to the x-axis and the y-axis. There is no such concept as a fixed background and variable background. This method can detect the moving objects by finding the change in position of pixels.

Optical flow method [15] is based on a relative movement, rather than the absolute motion, as in the case of motion vector search method. This technique uses the change trend of gray scale of each image point. Optical flow finds the image changes which are dependent on motion during a time interval. Optical flow field represents the speed of object movement of 3D points in a 2D image. However, this method is quite complicated and requires a higher material. So, as a consequence, it is not suitable for real time processing.

Space vector difference method [24] uses the difference in vector space to obtain a card unlike current video and the modeling of background frame. Then, an adaptive threshold is automatically calculated by analyzing the characteristic of difference map histogram. This method adopts the concept of technical RGB color. Space vector difference is applied to color images and gives good results, but it cannot detect the background objects if the color of the background and the object is the same.

2 Vehicles Detection

In proposed algorithm, Gaussian mixture model [23] is used to detect the foreground of video. There is need to provide some initial frames in Gaussian mixture model and then it computes probability density function (PDF) of each frame and then it measure the change in entrain intensity value of all frames. If change is greater than threshold intensity then value become part of foreground otherwise it assumes as background. A 0 or 1 is assigned to each pixel intensity value based on change in PDF in all initial frames. The concept of histogram equalization is observed to assign values. Fig. 3. shows the system flow diagram.

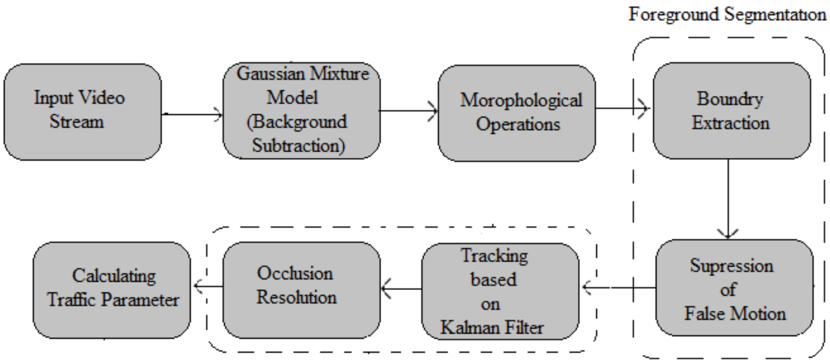


Fig. 3. Flowchart of Vision based Detection, Tracking

Gaussian Mixture model which is a background subtraction method is applied on a video stream shown in Fig. 4a, for foreground detection that is shown in Fig. 4b. In this Gaussian Mixture modeling, illumination variations in background are automatically adopted. The recent history $\{X_1, ..., X_t\}$ of a pixel is modeled through mixture of *K* (normally, 3 to 5) Gaussian distributions. The probability of current pixel is [23]:

$$P(X_t) = \sum_{i=1}^{K} w_{i,t} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \tag{3}$$

K defines the number of distributions, $\mu_{i,t}$ is the mean value of the i_{th} Gaussian in the mixture at time *t*, $w_{i,t}$ is an estimate of the weight of the i_{th} Gaussian in the mixture at time *t*, $\Sigma_{i,t}$ is the covariance matrix at time *t* for the i_{th} Gaussian in the mixture, and η is a Gaussian probability density function [23]:

$$\eta(X_t, \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} e^{-1/2(X_t - \mu_t)^T \Sigma^{-1} (X_t - \mu_t)} \tag{4}$$

If the current pixel value does not match with any of the *K* distributions than the least predictable distribution takes the place of a distribution that has current value as its mean value, low prior weight, and an initially high variance. The prior weights $\omega_{k,t}$ at time *t* for the *K* distributions [23]:

$$\omega_{k,t} = (1 - \alpha)\omega_{k,t-1} + \alpha(M_{k,t}) \tag{5}$$

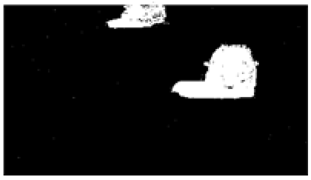
where $M_{k,t}$ is *'1'* for the matched model and *'0'* for the remaining models, and α is the learning rate. In addition, weights are renormalized after this approximation.

A surface is deemed to be background with higher probability (lower subscript *k*) if it occurs frequently and variation is not so much occurred. To set the background [23]:

$$\mathrm{B} = \arg\min_{b} \left(\sum_{k=1}^{b} w_k > T \right) \tag{6}$$

After detection of foreground by Gaussian Mixture model, next task is to remove noise in detected foreground. To this end, morphological operation is applied to get the best result. Morphological operation depends upon shape of the object. In this case, morphological closing with square structuring element provides best result. Same process is applied on all frames to detect noiseless foreground as shown in Fig. 4c. The next step is to boundary extraction and reduction of falsely detected objects (vehicle). For this purpose, a connected component analysis is performed and all objects that having area of less than 2500 are removed. Connected component analysis is applied on all detected vehicles.


(a)


(b)


(c)


(d)

Fig. 4 (a).Original Video Frame, (b).Detected Foreground, (c).Morphological Operation (d).Detected Vehicles

3 Vehicles Tracking and Occlusion Resolution

Once true objects are detected as shown in Fig. 5b then at next step, Kalman filter is applied for vehicle tracking and centroid of every detected vehicle is also determined. Tracking process can be disrupted if vehicles are occluded at time of entrance to frame. Generally, vehicles occlusion is seen on side by side in horizontal direction across adjacent lanes. To solve this issue, a prior occlusion detection method [25] is proposed which is based on lane information. Firstly, a lane mask *L(x, y)* is made with values:- *-1* (neglected area), *0* (separated area), *1* (first lane), and so on as shown in Fig. 5(b). A label *m* with a label image *f(x, y)* is assigned to each vehicle after connected-component labeling. After that, vehicles are monitored according to Eq.(7)

to get a histogram *G(m,h)* as regards to lane *h*. Eq.(8) is used with an adequate threshold *Thrsh = 5* to detect the occluded region. At the end, occlusion is removed through splitting the vehicles with the help of reference of *L(x, y).*

$$f(x,y) = m \text{ and } L(x,y) \neq -1 \text{ and } L(x,y) \neq 0$$
$$G(m, L(x,y)) = G(m, L(x,y)) + 1 \tag{7}$$
$$\text{if } \| \mathrm{G}(m,\mathrm{h}) - G(\mathrm{m},\mathrm{h}+1) \| < \text{Thrsh, then} \tag{8}$$

Resolve the occlusion

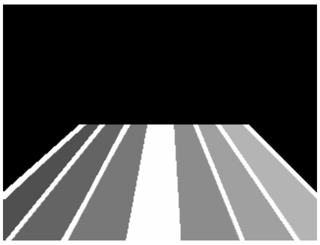


Fig. 5. Lane information in visual representation, gray-level regions = different lanes, white color regions = separation areas, black color regions = neglected area

In case, the aforementioned technique fails to handle the occlusion then a post occlusion detection method [25] is applied. The following steps are applied to determine that a vehicle is shaped by occluded vehicles or not.

1. If vehicle is not related to any existing trajectories then go to step 2. Otherwise, add the vehicle to trajectory.
2. If an offset which is calculated through centers of last two nodes of trajectory and vehicle region are a superset of last trajectory node then go to step 3. Otherwise, a new vehicle trajectory is created.
3. Object is split into two moving objects.

4 Traffic Parameters

A virtual line is drawn to count vehicle. When centroid touches virtual line, then algorithm checks the last five position of centroid and takes the decision for counting.
Algorithm keeps track of centroid value and stores center value of detected object for further computation. Traffic parameters are derived from tracking trajectories. However, in this proposed method, each trajectory is classified to a lane ID for calculating traffic parameters. The technique [25] which is used for classification a trajectory to lane ID is stated in Eq. (9). Way to get *G(m,h)* in Eq.(9) is the same as Eq.(7) and *l* is *m(n, t-1)* at time instance *(t-1)* which show the *n-th* trajectory.

$$h^*(\mathrm{n},t) = \arg\max_{h} G(m(\mathrm{n},t-1),h) \tag{9}$$

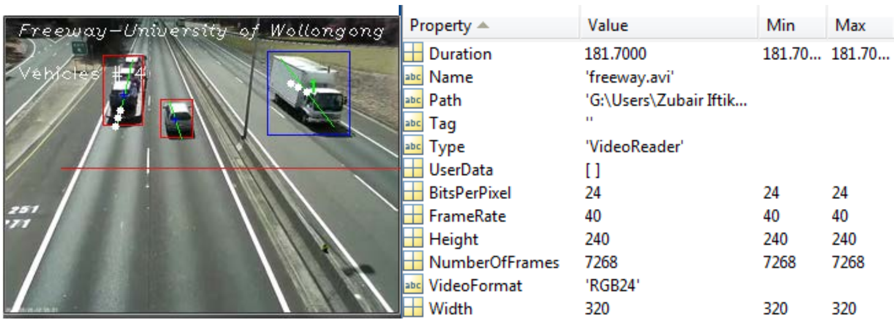
Following equations [25] listed in Table 1 are used to calculate the traffic parameters.

Traffic Parameters	Equations
Speed(h*)	$Speed(h^*(\mathrm{n,t})) = 1/8\,\mathrm{Speed(h^*(n,t))} + 7/8\,\frac{\lvert(C_{m(\mathrm{n,t-1})} - C_{m(n,t-N(n,t-1))}\rvert}{N(\mathrm{n,t-1})\,\mathrm{FramesperHour}}\frac{0.005}{\bar{P}(\mathrm{n})}$ N(n,t-1) is number of nodes at at time t-1in n-th trajectory , C is vehicle center, $\bar{P}(\mathrm{n})$ is nodes average width
Quantity(h*)	If N(n,t-1), Quantity (h*(n,t) = Quantity (h*(n,t) + 1
Volume (h*)	$Volume(h^*(n,t)) = \frac{\mathrm{Quantity(h^*(n,t))\,FramesperHour}}{t}$

Table 1. Traffic Parameters Equations

5 Experimental Results

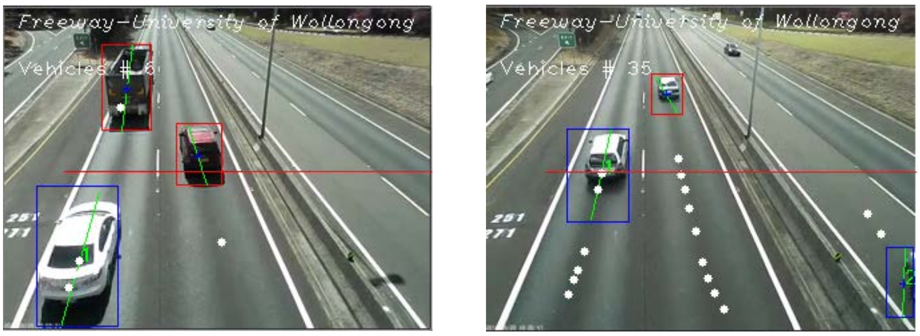
In these experiments, many common traffic video sequences are used which are available on different online data sources. However, we mostly focused on freeway traffic. In that, fixed position camera was located on a 1.5 meters long pole over a pedestrian bridge. Overall height of camera from the road is 6.5 meters, as observed in Fig. 6a. Experimental video was observed in a sunny day and consists of three minutes since *01:17pm* to *01:20 pm*.



(a) (b)

Fig. 6. (a) Counting and Tracking; a case of prior occlusion (b) Video Information

Information regarding to frame rate, data rate and siz of video can be seen in 6b. The proposed system is developed on Windows *7* platform with a Pentium® Dual-Core *2.1GHz CPU, 3GB Ram.*



(a) (b)

Fig. 7. (a) Speed 98.4km/hr, Quantity 06, Volume 1383/hr, the 1000-th Frame; (b) Speed 105km/hr, Quantity 35, Volume 1802/hr, the 5000-th Frame

4 Conclusion and Future Work

In this paper, different techniques of vehicle detection and tracking are discussed which are broadly categorized as sensor and vision based approaches. Researchers are taking interest in vision based approaches due to increasing quality of sensing device, computation power of computers, accuracy, human resource reduction and cost minimization. In proposed algorithm, Gaussian mixture model is used to detect foreground and Kalman filter is applied for tracking. In addition, a technique for occlusion handling is described. At the end, traffic parameters are determined. Future work is to enable system to handle serious occlusion and weather condition, and classification approach to differentiate the different types and sizes of vehicles.

References

1. P. Premaratne, S. Yang, Z. Zou, and P. Vial, “Australian Sign Language Recognition Using Moment Invariants”, Lecture Notes on Artificial Intelligence, vol. 7996, pp. 509–514, 2013.
2. P. Premaratne, S. Ajaz, and M. Premaratne, “Hand Gesture Tracking and Recognition System for Control of Consumer Electronics”, Springer Lecture Notes in Artificial Intelligence (LNAI), vol. 6839, pp 588-593, 2011.
3. P. Premaratne, Q. Nguyen & M. Premaratne, "Human computer interaction using hand gestures," in Advanced Intelligent Computing-Theories and Applications – Communications in Computer and Information Science, vol. 93, pp. 381-386, 2010.
4. P. Premaratne, F. Safaei and Q. Nguyen, Moment Invariant Based Control System Using Hand Gestures: Book Intelligent Computing in Signal Processing and Pattern recognition, Springer Berlin / Heidelberg, vol. 345, pp. 322-333, 2006.

5. P. Premaratne, M. Premaratne, Image Matching using Moment Invariants, Neurocomputing, Vol. 137, pp. 65-70, 2014.
6. P. Premaratne, S. Ajaz, M. Premaratne, "Hand Gesture Tracking and Recognition System Using Lucas-Kanade Algorithm for Control of Consumer Electronics", for Neurocomputing Journal, vol. 116(20), pp. 242–249, 2013.
7. P. Premaratne and Q. Nguyen, "Consumer electronics control system based on hand gesture moment invariants," IET Computer Vision, vol. 1, pp. 35-41, 2007.
8. S. Yang, P. Premaratne and P. Vial, Hand Gesture Recognition: An Overview, 5th IEEE International Conference on Broadband Network and Multimedia Technology, 2013.
9. Z. Zou, P. Premaratne, M. Premaratne, R. Monaragala, and N. Bandara, "Dynamic hand gesture recognition system using moment invariants", ICIAfS, IEEE Computational Intelligence Society, Colombo, Sri Lanka, pp. 108-113, 2010.
10. D.C. Herath, C. Kroos, C.J. Stevens, L. Cavedon, and P. Premaratne, "Thinking head: to-wards human centred robotic"', 2010 11th International Conference on Control, Automa-tion, Robotics and Vision (ICARCV), Singapore, pp. 2042-2047, 2010.
11. Minge, E. Evaluation of Non-Intrusive Technologies for Traffic Detection. Minnesota Department of Transportation, Office of Policy Analysis, Research and Innovation, SRF Consulting Group, US Department of Transportation, Federal Highway Administration, 2010.
12. S.-C. Cheung and C. Kamath, "Robust techniques for background subtraction in urban traffic video," in *Proc. EI-VCIP*, pp. 881–892, 2004.
13. B. Heisele, U. Kressel, and W. Ritter, "Tracking non-rigid, moving objects based on color cluster flow," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition, San Juan*, pp. 257–260, 1997
14. K. Housni, D. Mammass, and Y. Chahir, "Moving objects tracking in video by graph cuts and parameter motion model," *International Journal of Computer Applications*, 40(10):20-27, February 2012. Published by Foundation of Computer Science, New York, USA.
15. O. H. Ching, "Optical flow-based vehicle detection and Tracking," B.I.S. Thesis, University Tunku Abdul Rahman, Perak, Malaysia, 2010.
16. Setfano Messelodi, Carla Maria Modena, Nicola Sehata. "A Kalman Filter Based Background Updating Algorithm Robust to Sharp Illumination Change," *ICIAP 2005*. Italy, pp. 163-170, 2005.
17. Alatan A, Onural L, Wollborn M. "Image sequence analysis for emerging Interactive multimedia services: The European COST 211 framework," *IEEE Trans. Circuit Syst. Video Technol*, vol.8, no. 7, pp. 802-813, 1998.
18. L. Xiao, T. Li, "Research on Moving Object Detection and Tracking," FSKD 2010, vol. 2, pp. 2324-2327, 2010.
19. K. A. Ahmad, et al., "Improvement moving vehicle detection using RGB removal shadow segmentation," 2011, pp. 22-26.
20. D. Gangodkar, et al., "Robust segmentation of moving vehicles under complex outdoor conditions," IEEE Transactions on Intelligent Transportation Systems, vol. 13, pp. 1738-1752, 2012.

21. Bharti, T. Thind, "Background Subtraction Technique Review," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 2, no.3, pp. 166-168, 2013.
22. A. K. Chauhan, P. Krishan, "Moving Object Tracking Using Gaussian Mixture Model And Optical Flow," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 4, pp. 243-246, April 2013.
23. C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 246-252, 1999.
24. Z. Qian, et al., "Moving objects detection based on space vector difference," 2011, pp. 646-651.
25. S. P. Lin, et al., "A real-time multiple-vehicle detection and tracking system with prior occlusion detection and resolution, and prior queue detection and resolution," 2006, pp. 828-831.

[May 12, 2014]