2006 IEEE International Conference on
Systems, Man, and Cybernetics
October 8-11, 2006, Taipei, Taiwan

Traffic Monitoring and Vehicle Tracking using Roadside Cameras

Yao-Jan Wu, Feng-Li Lian, and Tang-Hsien Chang

***Abstract*—This paper studies the integration and implementation of digital image processing techniques on the roadside camera for traffic monitoring and vehicle tracking. The image processing framework developed in this study is mainly composed of five stages: (1) pre-processing, (2) foreground segmentation, (3) shadow removal, (4) tracking, and (5) traffic parameters extraction. During the pre-processing stage, the information of road geometry is obtained and the camera is calibrated. At the foreground segmentation stage and shadow removal stage, moving vehicles are segmented from the original input images. To make the system more robust, an α-β filter is used at the multi-vehicle tracking stage. Subsequently, related traffic parameters are extracted at the end of each tracking mechanism. The experimental results show that this system is capable of successfully extracting the traffic parameters, including the trajectory of the moving vehicles based on the image sequences captured by a digital camera on a free flow traffic in the daytime..**

I. Introduction

TRAFFIC Monitoring System (TMS) is mainly responsible for collecting, processing, analyzing, summarizing, and disseminating highway traffic data and is supported by a comprehensive computer database system. Hence, the first task of TMS is to produce reliable data that can be further analyzed manually or automatically in the Traffic Monitoring Center (TMC). With the collection of these traffic data, traffic engineers are able to design ultimate roadway geometry and sketch optimal traffic control.

Nowadays, it is common that local traffic information is presented in the format of video sequences captured without any special processing by the video camera system mounted on the pole near the roadside. Incidents locally can be detected through these video images and confirmed by human eyes directly. Furthermore, information obtained through video camera might provide the volume and occupancy data equivalent to those from multiple inductive loop detectors. Video and analyzed data are transmitted to a central location for further processing. They also provide a number of advantages over the traditional methods of traffic monitoring. In terms of microscopic detection, incidents, such as stopped or stationary vehicles, big change of vehicle speed, vehicles driving in the wrong direction, fallen objects, snaking vehicles, and stopped or moving vehicles on the road shoulder, can be confirmed in the detection zone or in the field of view of camera directly based on the microscopic traffic flow features acquired by image processing techniques. On the other hand, as for macroscopic detection, a larger set of traffic parameters such as spatial speed, occupancy, headway, queue length, etc., can be used to detect the incidents and further determine the type and location as well as the causes of congestion and recurrent accidents inside or outside the field of view.

Around the world, many computerized vision-based traffic monitoring systems (CVTMSs) have been done in past decades. Since 1993, U.C. Berkeley ROADWATCH Project had been focusing on a real-time computer vision system for measuring traffic parameters [1: Beymer et al. 1997], [2: Koller et al. 1994]. This project developed a feature-based tracking approach for the task of tracking vehicles under congestion. Kogut et al. [3: Kogut & Trivedi 2001] presented an algorithm that provides an example of such data fusion: maintaining the identity of a vehicle, or groups of vehicles as they pass through multiple cameras sites. Masoud et al. [4: Masoud et al. 2001] presented algorithms for vision-based monitoring of weaving sections. The proposed algorithms can track and count vehicles as they change lanes. Furthermore, they provide the velocity and the direction of each vehicle in the weaving section. Subsequently, Badenas et al. [5: Badenas et al. 2001] followed the previous research and further presented a region-based algorithm for segmenting and tracking moving objects in a traffic scene. Moreover, temporal information is integrated for improving frame-to-frame motion segmentation. Recently, Kumar et al. [6: Kumar et al. 2005] presented a real-time system for multi-target tracking and classification from the image sequences of a single stationary camera. Several targets can be tracked simultaneously in spite of splits and merges amongst the foreground objects and presence of clutter in the segmentation results.

In this study, the main objective of the proposed CVTMS is to utilize the digital image processing techniques to obtain fundamental traffic parameters in the field of view of the roadside camera. The proposed approach mainly consists of five stages: pre-processing, foreground segmentation, shadow removal, tracking, and traffic parameters extraction. The flow chart of proposed traffic monitoring system is

Yao-Jan Wu was with the Department of Civil Engineering, National Taiwan University, Taiwan. He is now with the Department of Civil Engineering, University of Washington - Seattle, USA (e-mail: yaojan@u.washington.edu).

Feng-Li Lian, was with. He is now with the Department of Electrical Engineering, National Taiwan University, Taiwan (corresponding author to provide phone: +886-2-3366-3606; fax: +886-2-23671909; e-mail: fengli@ntu.edu.tw).

Tang-Hsien Chang is with the Department of Civil Engineering, National Taiwan University, Taiwan.

This research was supported in part by the National Science Council, Taiwan, ROC, under the grants NSC 94-2218-E-002-064 and NSC 95-2218-E-002 -031.

shown in Fig. 1.

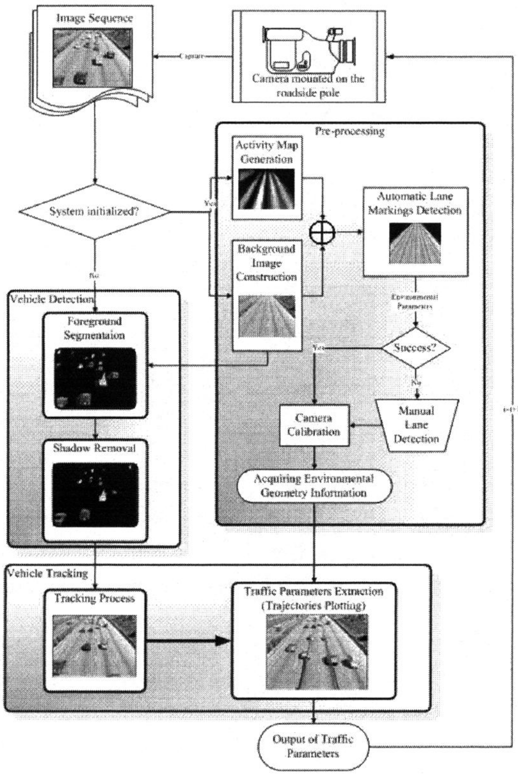


Figure 1: The flow chart of the proposed CVTMS.

II. PRE-PROCESSING

Pre-processing should be done in advance and contains the following four steps: background construction, activity map generation, lane detection and camera calibration. The procedure and results are elaborated in the following subsections and illustrated in Fig. 2.

A. Background image construction

The vehicle detection mechanism proposed in this study is based on the background subtraction method [7: Grupte et al. 2002], [8: Kilger 1992], [9: Veeraraghavan et al. 2003]. This method subtracts the input sequence from the background image to detect the changes in the environment and the difference images are further processed based on a pre-defined threshold to segment the moving vehicles from the background. The background image plays an important role in extracting moving foreground objects from video sequences. Herein, the progressive background image generation algorithm presented in [10: Chung et al. 2002] is adopted in this study for three main reasons. (1) low memory required. (2) fast background construction (less than 20 seconds during the non-peak hours). (3) update recursively. The updated background image is shown in Fig. 2(b)..

B. Preliminary lane detection

Since there is no prior information related to the scene layout obtained previously, the concept of activity map [11: Stewart et al. 1994] is adopted and revised to primarily identify the number of lanes in the field of view, and further provide crucial information for the following lane detection.

The idea of automatic lane detection is to accumulate a map of significant scene change that is mostly caused by the moving vehicles. There is a small accumulative value in the map of inactive area of no significant change in the intensity (e.g., areas around longitudinal lane markings). Hence the active area in the map is shown in high intensity, and inactive area in the map is in low intensity or even dark. If the difference of the pixels of two consecutive input images is higher than a pre-defined threshold, it can be regarded as a "significant change" and then these changes can be recorded as a binary image with adding one pixel value incrementally. The activity map is incremented by the binary difference image with noise removed by morphological operation, i.e., the dilation after the erosion. Finally, the activity map is done, as shown in Fig. 2(c).

C. Lane detections

On account of the unknown geometry of the road, it is necessary to detect the road area where the moving vehicles are going toward the same direction. The reason is that the vehicle detection will waste time to tracking the vehicles or other objects outside the detection area. On the other hand, if the lane information is obtained first, the camera calibration can be easily performed to construct the transformation between the real world coordinates and the image coordinates. Furthermore, the information of the road geometry is fairly important for verifying the outcome of vehicle detection and supporting the vehicle tracking. Herein, the proposed lane detection procedures are described as follows.

1) Average filtering with boxcar mask: The activity map should be analyzed to determine the number of lanes existing in the activity map. Once the activity map is formed, it is smoothed with a 3x3 boxcar mask, which is a low-pass filter to filter the 2-D signal of the activity map for the convenience of mountain-peak-n-valley-finding algorithm.

2) Mountain-peak-n-valley-finding algorithm: This algorithm is developed to trace the smoothed 1-D signal to locate the position of peak and valley individually in every y-interception of the filtered activity map and further distinguish the number of lanes by a voting manner.

3) Peak-value-finding algorithm: After analyzing the activity map, the obtained lane information is not sufficient to detect the lane markings. Hence, other features in the image should be involved to determine the actual lane markings. The peak-finding algorithm is developed to detect the lane markings in the front-view image captured by the camera mounting on the driving vehicle. Although the camera in this system is fixed on the static roadside pole, the proposed peak-value-finding algorithm adopts the aforementioned properties of the lane markings and moreover utilizes the idea of a moving window to determine the threshold for better result, as shown in Fig 3. The mountain-peak-n-valley map generated by this algorithm is shown in Fig.2(d).

4) Effective peak-value map: Because of the noise appearing outside the lane detection area, the peak-value map and mountain-peak-n-valley map are fused together to form an effective peak value map that only contains the points of lane markings without any interference. Actually, the lane markings in the background image are located within the area of two valleys of the mountain-peak-n-valley map. For this reason, the lane-markings-irrelevant peak value points can be removed by the mountain-peak-n-valley map and the effective peak-value map, as shown in Fig. 2(e). Utilizing the effective peak-value map can elevate success rate of the following lane detection procedure.

5) Hough transform [12: Gonzalez & Woods 2002]: The lane markings in the field of view are assumed lying on straight lines on the same plane and parallel to the same direction in the world coordinates. In the effective peak-value map, it is easily discovered that the individual set of lane markings are composed of several binary pixels of the value 1 (brightness). The global relationships between pixels are then constructed by utilizing the Hough Transform.

6) Manual lane detection: In case of the failure in automatic lane detection, a manual lane detection in the proposed system can be activated by the user. This procedure can be easily done using the hand held screen and a mouse at the camera site or remote computer.

D. *Camera calibration*

Camera calibration mainly determines the projection equations between the world coordinates and the image geometry. Bas and Chrisman [13: Bas & Chrisman 1997] proposed an easy way to install camera calibration for traffic monitoring.

With the relation established after calibration, the top-view image imitates the aerial photo captured by the camera elevated certain meters high and moved certain meters forward, as shown in Fig. 2(h). Most important of all, the lane width in real world is known – usually 3.6 meters in the highway. Hence we can know that every pixel in the top-view image equivalent of the real distance in the world and the vehicle speed can be further estimated.

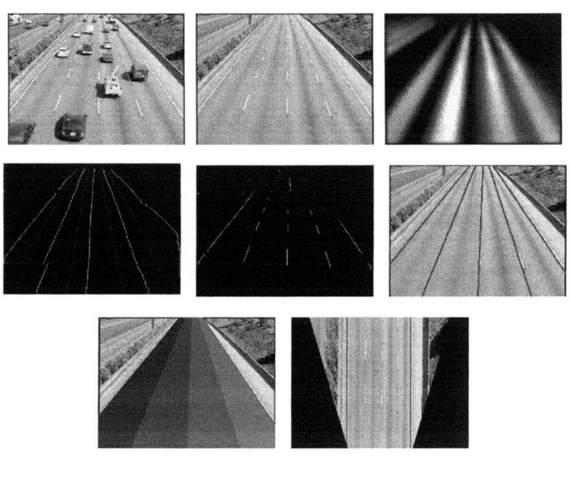


Figure 2: Pre-processing, (a) the input sequence, (b) the constructed background, (c) activity map (gray-level scaled for visualization), (d) peak-n-valley map, (e) effective peak-value map, (f) the result of the automatic lane detection- the result of Hough Transform, (g) road segmentation, (h) top-view image – after camera calibration.

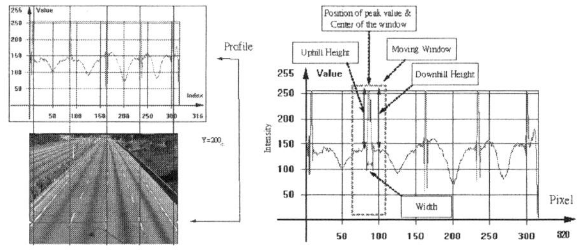


Figure 3: Peak-value-finding.

III. VEHICLE DETECTION

The proposed method of vehicle detection consists of two major steps: foreground segmentation and shadow removal. In the foreground segmentation step, the moving foreground objects, possibly vehicles, are extracted from the foreground image and expressed in the format of a binary moving foreground mask (MFM). However, in some cases, the MFM might cover the region of moving vehicles adjoining their moving shadow. The region of adjoining shadow is then eliminated from the MFM in the second step shadow removal. Finally, a binary moving object mask (MOM) standing for the moving vehicle without containing the region of the shadow is outputted for the following tracking stage described in Section IV.

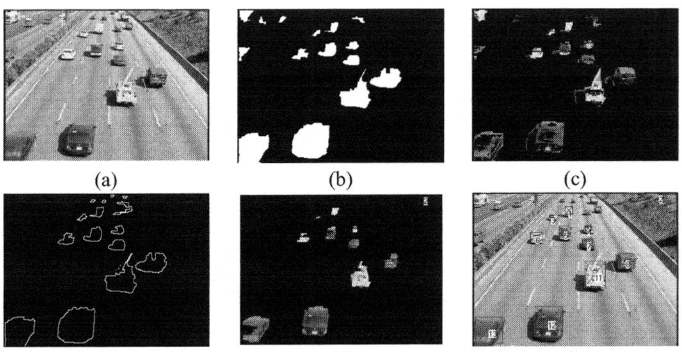
A. *Foreground segmentation*

Foreground segmentation is the major part of vehicle detection. If the sunlight is strong, the accuracy of the shadow removal procedure primarily depends on the outcome of the foreground segmentation. Although the background subtraction method is used to perfectly extract the moving foreground of the input sequences (see Fig. 4(a)), the dilation operation after the erosion operations and region analysis techniques are employed to remove some unreasonable foreground objects made by noise. After the hole-filling algorithm is done, the binary MFM is the final result as the input to the shadow removal process or tracking process directly, as shown in Fig. 4(b).

B. *Shadow removal*

Moving shadows would cause problems on the correctness of the localization and the measurements and detection of moving objects. In previous subsection, all the moving points of both objects and shadows are detected at the same time. Moreover, shadow points are usually connected to objects. The objective here is to detect the cast shadow from the object, and to remove only cast shadow from every MFM.

The first step in the proposed shadow removal method is to eliminate the shadow hypotheses and all the pixels in the moving objects are analyzed by employing the characteristics of difference color spaces. The pixels on the moving foreground are further classified as the foreground class and the shadow hypothesis class by one scoring method. Later on, the true vehicle object without shadow is re-constructed in the vehicle object recovery step by the edge information and brightness. Edge information can be deemed as the illumination invariant descriptor for distinguishing the moving shadow from the moving vehicles [14: Wang et al. 2004]. The edge feature map can be obtained by using the Sobel edge detector that has sufficient edge information to make up for the lost region of one moving object. Bright foreground pixels are impossible classified as shadows. As a result, a foreground pixel in the moving mask objects is regarded as a bright pixel if any one of its RGB values is higher than that of its corresponding background pixel. After padding the lost region and eliminating the contour of the MFM (as shown in Fig. 4(d)), the process outputs the moving foreground image corresponding to moving object mask (MOM), as shown in Fig. 4(e). Moreover, each of the vehicles in the foreground image are enclosed by a bounding box corresponding to the moving foreground image and each one is assigned a unique and time-independent label number for further tracking process, as shown in Fig. 4(f). Note that all the labels are stored in a list to ensure their existence.



(a) (b) (c)

(d) (e) (f)

Figure 4: Procedures of vehicle detection.

IV. VEHICLE TRACKING

In this section, a vehicle tracking method that consists of 3 stages: region splitting, region grouping and tracking, is discussed.

A. Region splitting

Because some input MOMs may be connected to each other in some cases. The vehicles are regarded as a big moving foreground when two MOMs of different vehicles are connected to each other,, a region splitting is adopted to separate one region into two possible regions of the isolated vehicles. At first, the definition of oversize object is determined according to the width of a lane. If an oversize object is decided, the projection of the MOM is calculated. Furthermore, the location of the least accumulation is found. Finally, the region is separated into two regions assigned by two distinct newly numbers, respectively.

B. Region Grouping

Unexpected scattered regions sometimes happen as a result of poor vehicle detection caused by the change of light, camera aperture, or noises. Gupte et al. [7: Gupte et al. 2002] proposed a vehicle tracking method that formed an association graph between the region from the previous frame and the region in the current frame. However, with a better foreground segmentation, it is not necessary to track multiple regions in the time domain. Thus, we simply treat the scattered regions in the spatial domain by some grouping criteria and constraints that can combine this scattered regions based on their rectangular bounding box on the spatial domain.

After processed by the grouping procedure, almost all the scattered regions can be combined into one complete bounding box that represents a model of the vehicle..

C. Tracking

Tracking is one of the most important tasks in the traffic monitoring system. With good tracking techniques, some false alarms caused by the poor vehicle detection can be neglected. Thus, a tracking procedure that associates the bounding regions existing in the current frame with the bounding regions existing in the previous frame is proposed. In this section, a tracking relationship table that can help the tracking procedure is first presented. The data prediction and smooth mechanism is used for obtaining a better tracking result. Finally, the tracking result is illustrated as the trajectory lines traveled by the identified vehicles.

1) *Construct tracking relationship table*: As shown in Fig. 4(f), every vehicle was marked by a rectangle. In different time stamp, every vehicle is assigned a newly labeled number. Hence, the corresponding relationship between two closest points with a distance less than a pre-defined threshold in different time should be found since we assume that a vehicle dose not move too fast within a short time interval. As illustrated in Fig. 5, these relationships are connected to form a table that makes the tracking process operate easily. Each block stands for a spatial- and time-independent vehicle at the different time frame.

2) *Initiate a tracking:* If the labeled vehicle in the current image cannot find the corresponding number in the previous image, the number labeling for the vehicle is initiated.

3) *End a tracking:* If there is no vehicle number corresponding to any vehicle number in the previous frame, the vehicle number in the previous frame is held on until any vehicle number in the following frames is found. However, this hold-on procedure is limited to n times or this tracking is ended. During the hold-on procedure, the prediction by the α-β filter is actuated.

4) *Vehicle motion prediction*: The tracking might not be stable since the observed data could contain some noises or even be missing. Hence, the Kalman filtering techniques for the recursive target state estimation is widespread adopted in some target tracking related researches [9: Veeraraghavan et al. 2003], [15: Lin et al 2003]. However, using the Kalman filter suffers from the drawback of complicated calculation and high processing time. As a result, a fixed-coefficient

filtering technique such as the α-β filter and α-β-γ filter [16: Blackman 1986] is adopted to perform similar work as the Kalman filter. The equation related to γ in the α-β-γ filter is quite sensitive to observation data. The α-β filter is used to smooth the observed data and improve the tracking performance. The filter is defined as the following equations:

$$x_s(k) = \hat{x}(k \mid k) = x_p(k) + \alpha_x \left[x_o(k) - x_p(k) \right] \quad (1)$$

$$y_s(k) = \hat{y}(k \mid k) = y_p(k) + \alpha_y \left[y_o(k) - y_p(k) \right] \quad (2)$$

$$v_{sx}(k) = \hat{\dot{x}}(k \mid k) = v_{sx}(k-1) + \frac{\beta_x}{qT} \left[x_o(k) - x_p(k) \right] \quad (3)$$

$$v_{sy}(k) = \hat{\dot{y}}(k \mid k) = v_{sy}(k-1) + \frac{\beta_y}{qT} \left[y_o(k) - y_p(k) \right] \quad (4)$$

$$x_p(k+1) = \hat{x}(k+1 \mid k) = x_s(k) + T \cdot v_{sx}(k) \quad (5)$$

$$y_p(k+1) = \hat{y}(k+1 \mid k) = y_s(k) + T \cdot v_{sy}(k) \quad (6)$$

where $x_o(k)$ and $y_o(k)$ are the observed target positions in the x and y direction, respectively, at the k_{th} scan, $x_p(k)$ and $y_p(k)$ are the predicted target positions in the x and y direction, respectively, at the k_{th} scan, $x_s(k)$ and $y_s(k)$ are the smoothed target positions in the x and y direction, respectively, at the k_{th} scan, $v_{sx}(k)$ and $v_{sy}(k)$ are the smoothed target velocity in the x and y direction, respectively, at the k_{th} scan, T is the sampling interval, q is the number of scans since the last measurement;, and α_x, β_x, α_y, β_y are the fixed-coefficient filter parameters in the x and y directions, respectively.

5) *Trajectory plotting:* After the relationship table is constructed, the trajectory of each identified vehicle is sketched and updated with its moving state.

D. *Traffic Parameters Extraction*

Key traffic parameters include flow, speed, density, etc., which can be observed and studied at the microscopic and macroscopic levels. The definitions of theses parameters extracted in the system can be found in [17: May 1990]. As illustrated in Fig. 6(a), the detection zone is confined within the upper and lower bound defined in advance. The measurement of the speed and the counting of one individual vehicle are finished at the upper bound of the detection zone. The extracted data are the output on the display panel, as shown in Fig. 6(b).

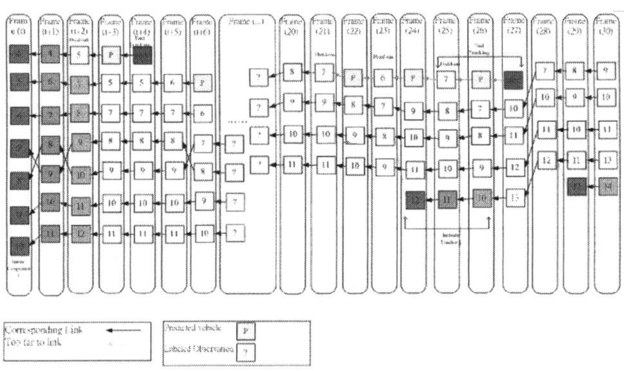
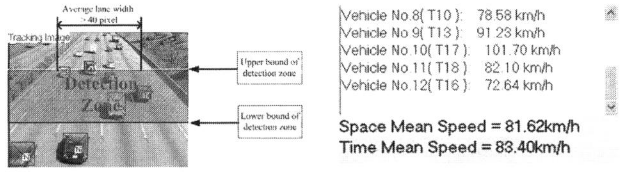


Figure 5: Tracking Relation Table.



(a) (b)

Figure 6: Traffic parameters extraction, (a) Detection Zone, (b) The outputs.

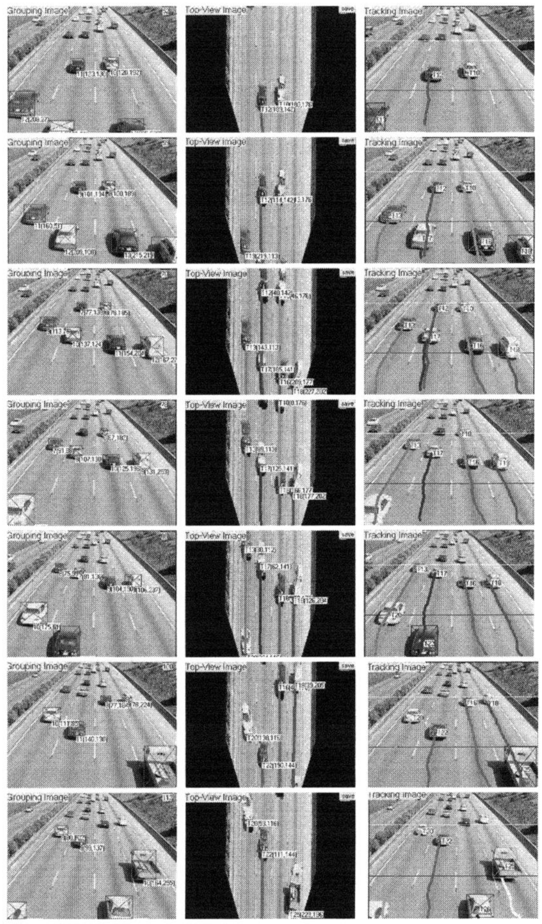


Figure 7: Result of tracking and trajectory plotting.

V. EXPERIMENTAL RESULT

The input image sequences are captured by a CCD camera mounted on the bridge at the freeway interchange. The camera overlooks the highway traffic flow moving in either south or north directions. The size of the image sequences are 320 (width) X 240 (height) in pixels and the frame rate is set at 30 frames per second (fps). This system is developed by the Borland® C++ Builder based on Microsoft® Windows operating system.

The profile of the test clips and results are shown in Table 1. Traffic parameters, such as volume, speed, and density, are extracted from these five test clips. Comparing to the actual volume collected by the system, this tracking system performed well with few errors in the test sequences.

Moreover, the trajectory of each vehicle is constructed successfully even it is tolerably influenced by the false alarms coming from the vehicle detection. However, it can be found that the error rate of the vehicle count (traffic volume) in Sequences 3 and 5 are higher. The reason is that some of moving vehicles are clustered and the occluded to each others frequently. Importantly, the average success rate of vehicle counting is higher than 96 %.

VI. SUMMARY

A computer vision-based traffic monitoring system is developed in this paper. The automatic lane detection and manual lane detection are developed to acquire geometry parameters in the pre-processing stage. The automatic lane detection is convenient for users to set up the camera promptly. The mountain peak-n-valley-finding algorithm, peak-value-finding algorithm and Hough transform are well-cooperated for finding the lane boundaries. In the foreground segmentation, the result showed that the algorithm indeed successfully extracted moving vehicles with shadow removing. At the vehicle tracking stage, the $\alpha - \beta$ filter is adopted that makes the system performance more robust. The experimental results show that the average success rate of vehicle counting is higher than 96 % on free flow traffic with a clear day. For future studies, it is recommended to include some more scenarios, e.g. bad weather conditions and afternoon peak hours).

REFERENCES

[1] D. Beymer, P. McLauchlan, B. Coifman, and J. Malik, "A Real-time Computer Vision System for Measuring Traffic Parameters," in *Proc. IEEE Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, pp. 495-501, June, 1997.

[2] D. Koller, J. Weber, and J. Malik, "Robust Multiple Car Tracking with Occlusion Reasoning," In *Proc. Third European Conference on Computer Vision*, pp. 186-196, May 2-6, 1994.

[3] G. Kogut, and M. M. Trivedi, "Maintaining the Identity of Multiple Vehicles as They Travel Through a Video Network," in *Proc. IEEE Int'l Conf. on Intelligent Transportation Systems*, Oakland, California, Aug. 2001.

[4] O. Masoud, N. P. Papanikolopoulos, and E. Kwon, "The use of computer vision in monitoring weaving sections," *IEEE Transactions on Intelligent Transportation Systems*, vol. 2, no. 1, pp. 18-25, Mar. 2001.

[5] J. Badenas, J. M. Sanchiz, and F. Pla, "Motion-based segmentation and region tracking in image sequence," *Pattern Recognition*, vol. 34, no. 3, pp. 661-670, Mar. 2001.

[6] P. Kumar, S. Ranganath, W. Huang, and K. Sengupta, "Framework for Real-time Behavior Interpretation from Traffic Video," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 1, pp. 43-53, Mar. 2005.

[7] S. Gupte, O. Masoud, R. F. K. Martin, and N. P. Papanikolopoulos, "Detection and Classification of Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 3, no. 1, pp. 37-47, Mar. 2002.

[8] M. Kilger, "A Shadow Handler in a Video-based Real-time Rraffic Monitoring System," in *Proc. IEEE Workshop on Applications of Computer Vision*, pp. 11-18, Nov. 1992.

[9] H. Veeraraghavan, O. Masoud, and N. P. Papanikolopoulos, "Computer Vision Algorithms for Intersection Monitoring," *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, no. 2, pp. 78-89, June 2003.

[10] Y. C. Chung, J. M. Wang, and S. W. Chen, "Progressive Background Images Generation," in *Proc. the 15th IPPR Conference on Computer Vision, Graphics and Image Processing*, Hsinchu, Taiwan, Aug. 2002.

[11] B. D. Stewart, I. Reading, M. S. Thomson, T. D. Binnie, K. W. Dickinson, and C. L. Wan, "Adaptive lane finding in road traffic image analysis," in *Proc. Seventh International Conference on Road Traffic Monitoring and Control*, pp.133 -136, Apr. 1994.

[12] R. C. Gonzalez, and R. E. Woods, *Digital Image Processing*, 2nd Edition, Prentice Hall, 2002.

[13] E. K. Bas, and J.D. Crisman, "An Easy to Install Camera Calibration For Traffic Monitoring," *in Proc. IEEE Conference on Intelligent Transportation Systems*, pp.362-366, 1997

[14] J. M. Wang, Y. C. Chung, C. L. Chang, and S.W. Chen, "Shadow Detection and Removal for Traffic Images," *in Proc. the 2004 International Conference on Networking, Sensing & Control*, Taipei, Taiwan, Mar. 2004.

[15] C.-P. Lin, J.-C. Tai, and K.-T. Song, "Traffic Monitoring Based on Real-Time Image Tracking," in *Proceedings of the 2003 IEEE International Conference on Robotics & Automation*, pp. 2019-2096, Sep. 2003.

[16] S. S. Blackman, *Multiple-Target Tracking with Radar Applications*, Norwood, MA: Artech House, 1986.

[17] A. D. May, *Traffic Flow Fundamentals*, *Prentice Hall*, 1990.

TABLE I
EXPERIMENTAL RESULTS

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
Clip 1	**4**	**1500**	**86**	**87**	**1.15**	**645.00**	**89.16**	**88.13**	**17.56**
Clip 2	**4**	**1500**	**54**	**54**	**0**	**833.33**	**95.00**	**93.48**	**8.82**
Clip 3	**4**	**1500**	**92**	**98**	**7.14**	**1548.0**	**86.96**	**86.49**	**17.90**
Clip 4	**5**	**2500**	**114**	**115**	**0.87**	**984.99**	**85.58**	**82.81**	**11.89**
Clip 5	**5**	**1900**	**82**	**89**	**7.90**	**932.26**	**87.00**	**85.59**	**10.89**
Total			**428**	**443**	**3.40**	**Success rate= 96.6%**			

(1) Number of lanes
(2) Total Frames
(3) Detected volume: The total number of vehicles passing the upper bound of the detection zone is collected by the tracking system.
(4) Actual volume: The total number of vehicles passing the upper bound of the detection zone is collected by manual counting.
(5) Error rate (%): (Actual volume – Detected volume)/ Actual volume.
(6) Flow rate: vehicles per hour per lane
(7) TMS (Time Mean Speed): kilometer per hour.
(8) SMS (Space Mean Speed): kilometer per hour.
(9) Density: vehicle per kilometer per lane.
Annotation: the figures marked by mean the measurement.