in *Proc. IEEE Int. Conf. Syst., Man, Cybern. Comput. Cybern. Simul.*, Oct. 12–15, 1997, vol. 1, pp. 135–140.
[5] D. Xue and Z. Dong, "An intelligent contraflow control method for real-time optimal traffic scheduling using artificial neural network, fuzzy pattern recognition, and optimization," *IEEE Trans. Control Syst. Technol.*, vol. 8, no. 1, pp. 183–191, Jan. 2000.
[6] S. Kim and S. Shekhar, "Contraflow network reconfiguration for evacuation planning a summary of results," in *Proc. GIS*, Bremen, Germany, Nov. 2005, pp. 250–259.
[7] S. Kim, S. Shekhar, and M. Min, "Contraflow transportation network reconfiguration for evacuation route planning," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 8, pp. 1115–1129, Aug. 2008.
[8] J. W. Wang, H. F. Wang, W. J. Zhang, and K. Furuta, "On an integrated approach to emergency evacuation and resilience for transportation service systems," *IEEE Syst. J.*, in Press.
[9] J. W. Wang, W. H. Ip, and W. J. Zhang, "An integrated road construction and resource planning approach to the evacuation of victims from single source to multiple destinations," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 277–289, Jun. 2010.
[10] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw., Perth, Australia*, Nov./Dec. 1995, vol. 4, pp. 1942–1948.
[11] J. W. Wang and D. W. Wang, "Particle swarm optimization with a leader and followers," *Progr. Nat. Sci.*, vol. 18, no. 11, pp. 1437–1443, Nov. 2008.
[12] H. W. Hamacher and S. A. Tjandra, "Mathematical modelling of evacuation problems: A state of art," in *Pedestriori and Evacuation Dynamics*, M. Schreckcnberg and S. D. Sharma, Eds. New York: Springer-Verlag, 2002, pp. 227–266.
[13] J. W. Wang, F. Gao, and W. H. Ip, "Measurement of resilience and its application to enterprise information systems," *Enterprise Inf. Syst.*, vol. 4, no. 2, pp. 215–223, May 2010.
[14] W. J. Zhang and Y. Lin, "On the principle of design of resilient systems-application to enterprise information systems," *Enterprise Inf. Syst.*, vol. 4, no. 2, pp. 99–110, May 2010.
[15] G. L. Hamza-Lup, K. A. Hua, L. Minh, and P. Rui, "Dynamic plan generation and real-time management techniques for traffic evacuation," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 4, pp. 615–624, Dec. 2008.
[16] Y. Chiu and P. B. Mirchandani, "Online behavior-robust feedback information routing strategy for mass evacuation," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 2, pp. 264–274, Jun. 2008.
[17] A. Shende, M. P. Singh, and P. Kachroo, "Optimization-based feedback control for pedestrian evacuation from an exit corridor," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1167–1176, Dec. 2011.
[18] M. Di Gangi, "Modeling evacuation of a transport system: Application of a multimodal mesoscopic dynamic traffic assignment model," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1157–1166, Dec. 2011.
[19] S. A. Wadoo and P. Kachroo, "Feedback control of crowd evacuation in one dimension," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 1, pp. 182–193, Mar. 2010.
[20] S. Xu and H. B.-L. Duh, "A simulation of bonding effects and their impacts on pedestrian dynamics," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 1, pp. 153–161, Mar. 2010.

# A New Approach to Video-Based Traffic Surveillance Using Fuzzy Hybrid Information Inference Mechanism

Bing-Fei Wu, *Fellow, IEEE*, Chih-Chung Kao, *Member, IEEE*, Jhy-Hong Juang, and Yi-Shiun Huang

***Abstract*—This study proposes a new approach to video-based traffic surveillance using a fuzzy hybrid information inference mechanism (FHIIM). The three major contributions of the proposed approach are background updating, vehicle detection with block-based segmentation, and vehicle tracking with error compensation. During background updating, small-range updating is adopted to overcome environmental changes under congested conditions. During vehicle detection, the proposed approach detects the vehicle candidates from the foreground image, and it resolves problems such as headlight effects. The tracking technique is employed to track vehicles in consecutive frames. First, the method detects edge features in congested scenes. Next, FHIIM is employed to determine the tracked vehicles. Finally, a method that compensates for error cases under congested conditions is applied to refine the tracking qualities. In our experiments, we tested scenarios both inside and outside the tunnel with three lanes. The results showed that the proposed system exhibits good performance under congested conditions.**

***Index Terms*—Congested condition, traffic surveillance, vehicle detection, vehicle tracking.**

## I. Introduction

The use of vehicles for transport is rapidly increasing with improvement in our quality of life. However, such an increase in the use of vehicles compounds traffic problems. Therefore, intelligent transportation systems have become very popular research fields. To monitor traffic, vision-based traffic surveillance is one of the most popular methods, and comprehensive and up-to-date surveys are provided in [1] and [2].

Although many surveillance methods have been presented, two problems still need to be solved in congested situations, especially inside tunnels. First, the low angle of the camera means that vehicles are easily connected visually in images. Therefore, occluded vehicles need to be separated to achieve detection accuracy. Second, the background updating in congestion is important because of the cover of vehicles. In this brief, we propose three methods for resolving these problems. First, a modified background-updating procedure is proposed to smoothly update a background image during congestion. Second, occluded vehicles are segmented based on their interinformation. Third, tracking verification is performed by our proposed fuzzy hybrid information inference mechanism (FHIIM), where compensation is invoked when the comparison result is dissimilar.

This brief is organized as follows: Section II describes related works. Section III describes the background extraction and updating procedure. Vehicle extraction and vehicle tracking are addressed in Sections IV and V. The experimental results are presented in Section VI, and finally, our conclusions are stated in Section VII.

Manuscript received January 19, 2012; revised April 16, 2012; accepted June 19, 2012. Date of publication August 9, 2012; date of current version February 25, 2013. This work was supported by the National Science Council, Taiwan, under Grant NSC 100-2221-E-009-041. The Associate Editor for this paper was J. Zhang.

The authors are with the Department of Electrical and Control Engineering, National Chiao Tung University, Hsinchu 300, Taiwan (e-mail: kevinkao@cssp.cn.nctu.edu.tw).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2012.2208190

## II. Related Works

Several researchers developed great systems for traffic surveillance over the past few decades. Kanhere and Birchfield [3] presented a taxonomy for roadside camera calibration. Wang [4] presented an overview of the background, concepts, basic methods, major issues, and current applications of parallel transportation management systems. Tsai *et al.* [5] presented a novel vehicle detection approach for detecting vehicles based on static images that used color and edges. This method introduced a new color transform model for finding important "vehicle colors" to quickly locate possible vehicle candidates. Kanhere and Birchfield [6] provided a method for segmenting and tracking vehicles on highways using a camera that was of relatively low level. Wang [7] proposed a joint random field model for moving vehicle detection in video sequences. The proposed method could handle moving shadows, lights, and various weather conditions. Melo *et al.* [8] described a low-level object-tracking system that produced accurate vehicle motion trajectories, which could be further analyzed to detect lane centers and classify lane types. Mosabbeb *et al.* [9] used shadows as a feature of vehicles, and a background model verified the detection of shadows. In addition to daytime traffic monitoring, tracking and pairing of vehicle headlight was also addressed in [10] and [11]. A self-diagnosing intelligent surveillance system is designed for monitoring both daytime and nighttime scenes [12]. Background modeling is used in the following video surveillance systems. However, the update in congested flow is less addressed. Self-adaptive background subtraction was proposed to refine the background [13]. Kumar *et al.* [14] segmented the vehicles from the background and classified the vehicle using Bayesian network. Hsieh *et al.* [15] identified lane locations and their corresponding lane widths, and then, line-based shadow elimination was performed to separate the occlusion caused by shadows. Pang *et al.* [16] proposed a generalized deformable model for segmenting the occlusion blobs in the foreground image.

## III. Background Extraction and Updating

Background model is one of the most useful methods for change detection. Therefore, many background-modeling methods were used in previous studies, such as sigma–delta estimation (SDE) [17] and the Gaussian Mixture Model.

Generally speaking, the color intensities gradually change in the monitored environment, and the background image should be smoothly updated to overcome luminance changes. Therefore, a normal updating using the SDE method can successfully adjust the colors of the reference background image in stable traffic flows. However, under congested conditions, many vehicles cover the background pixels for a long period. The updating problems become difficult, because the background image needs to be accurately updated and it also needs to simultaneously detect moving objects. Therefore, the occupancy $(K)$ in (1) is computed to distinguish whether the flow is free or forced. If the traffic flow is free, a normal updating flow is performed. Otherwise, our proposed small range updating is applied to update the background image

$$K = \frac{N_i}{N} \times 100\% \tag{1}$$

where $N$ is the number of frames in a period, and $N_i$ is the number of frames where the lane is occupied by vehicles. An occupancy example is shown in Fig. 1.

Under the congested conditions shown in Fig. 2, SDE may lead to false updates of nonbackground information in the background model. As a result, an approach that can be applied to solve this serious problem is proposed here. In congestion, the background intensities are assumed to change within a certain small range. The gray levels

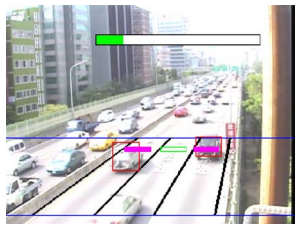


Fig. 1. Example of occupancy.

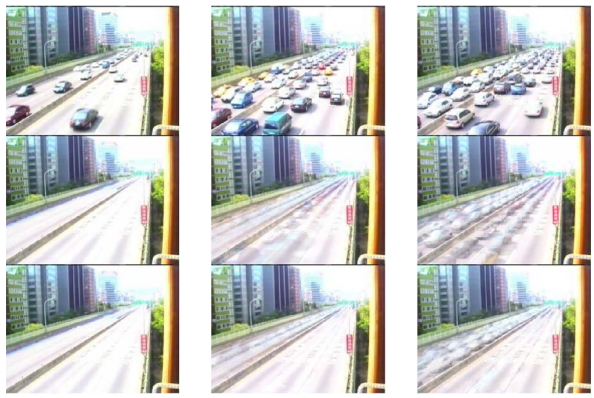


Fig. 2. Experimental results after applying the method to congested flows. (Top) Input image. (Middle) Updated background image using SDE method. (Bottom) Updated background image using the small-range update.

between 0 and 255 are equally divided into $N$ small ranges, where $N$ denotes the number of ranges. The histogram statistics for each pixel then only count the number in each small range. The statistic value $S_n(m)$ of the $m$th pixel in the $n$th small range is calculated using (2), shown below, whereas the average gray value $\mu_n(m)$ in the $n$th small range is calculated using (3), also shown below,

$$S_n(m) = \begin{cases} S_n(m)+1, & \text{if } I(m) \in \left[\frac{n\times 256}{N}, \frac{(n+1)\times 256}{N}\right) \\ S_n(m), & \text{otherwise} \end{cases} \tag{2}$$

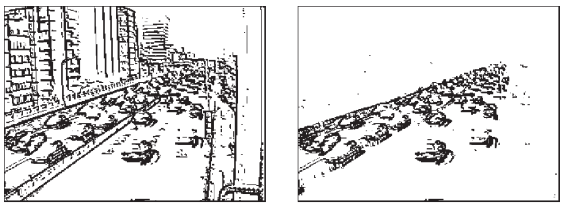
$$\mu_n(m) = \frac{1}{S_n(m)} \sum_{i=0}^{S_n(m)-1} I_i(m) \tag{3}$$

where the intensity of the $m$th pixel in the current frame is $I(m)$, and $I_i(m)$ is the $i$th intensity in the $n$th range.

After computing the average value $\mu_n$ in each small range, the background model can be determined as updated or not. To avoid updating the foreground object into the background image, the updating should be very carefully performed in a small range. After the absolute difference value of the background intensity and the average value $\mu_n$ in each range is obtained, the background intensity is updated if the difference is smaller than the range. Otherwise, this background pixel is not updated. By evaluating different scenarios and analyzing different experimental results, the value of $N$ is 16, whereas the period for accumulating the number of frames in the $n$th range is 150 frames. If the occupied frame number is more than 90, the small range update is performed for the congested condition. Fig. 2 shows the experimental results after applying the method under congested conditions. Compared with updating using the traditional SDE method, the side effects of moving vehicles in congestion are significantly better removed by applying the proposed updating method.

## IV. Vehicle Extraction

In this section, we introduce a method for separating moving objects from an input image. First, foreground segmentation and moving-edge



(a) (b)

Fig. 3. Example of moving-edge detection. The noise edges generated by road marks and lane marks are greatly reduced. (a) Original Sobel edge. (b) Moving edge.

extraction are used to obtain the moving features of vehicles. Then, connected component labeling is used to identify the shapes of objects. Finally, each labeled object is verified based on its visual features. Block-based foreground segmentation is adopted here to decrease the noise of pixels, and it also reduces the computation loading by applying connected component labeling. We suppose that the size of the input image is $M \times N$, and the size of a designated block is $D \times D$. The input image can then be separated into $M_B \times N_B$ blocks. For each block, the sum of the absolute difference (SAD) is calculated between each input image and the background image to confirm whether each block is foreground or not. Connected component labeling is then applied to identify the shapes of the moving blocks. To resolve the problem of visually connected vehicles, the information from lanes is used to mask the segmentation images, so false-merged objects can be successfully separated. Although tall vehicle may segment into two vehicles, the edge and tracking is applied to verify. After connected component labeling, the rectangle information is obtained. However, even the block-based segmentation is performed, some tiny objects generated by noises whose visual widths is less than half the lane width should be removed. In addition to the foreground, the moving edge is an important feature of the proposed system. The advantage of applying edge is their stability under different luminance conditions. Moreover, the moving edge can vertically separate two vehicles. Edge features are obtained by applying a Sobel operator to the image. However, the edges produced by lane marks and road marks may have an impact on the shapes of vehicle edges. Thus, the edges of a background image are taken into consideration to reduce the noise. Therefore, the edge generated by the background, lane marks and road marks can be removed using the edge in the background image. The comparison results are shown in Fig. 3.

## V. Vehicle Tracking

A tracking procedure is required by the surveillance system to identify the same vehicle in consecutive frames. Objects may be strongly connected into one large rectangle after connected component labeling. Therefore, measurements in the current frame should consider the information from the previous frame, and the tracked object should be masked to avoid effects on the new tracking. A Kalman filter has been successfully used for moving object tracking. However, the measurement noise for width and height of vehicles in different positions is hard to measure. In this brief, a new procedure was developed for simultaneously tracking multiple vehicles, particularly under congested conditions. First, the new positions of the tracked objects are measured using a feature-searching method. Second, the FHIIM method is used to verify the tracking results and to decide whether to compensate for the errors. When the result needs to be compensated, a prediction is used to refine the result. Finally, masks are used to remove the tracked objects from the foreground images.

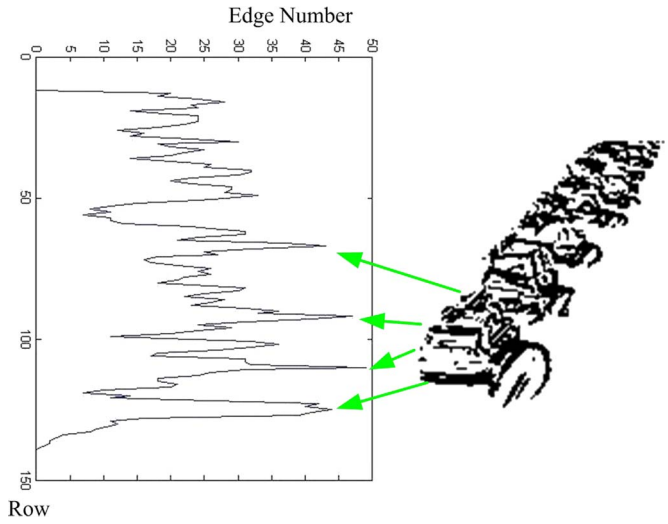



Fig. 4. Vehicle edge vertical projection under congested conditions.

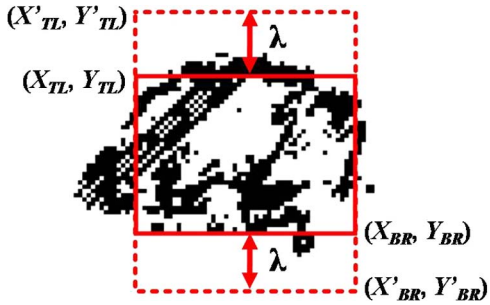



Fig. 5. Diagram of the ROI extension for a moving-edge image.

### A. Searching for the Tracked Vehicles

Under congested conditions, incorrect vehicle boundaries may be detected due to the visually connected vehicles in the foreground image. To resolve this problem, the moving edge is used to determine the top and bottom boundaries of vehicles. The moving-edge feature is the most robust feature of vehicles, and they have few distortions under congested conditions. Fig. 4 shows the edge image in the middle lane and the corresponding vertical projection under congested conditions. In this figure, the edges still maintain their properties of a local maximum in the vertical projection. This means that the edges can be used to identify vehicle boundaries under congested conditions. The detailed steps of boundary searching are described here.

Step 1: Set the extension. The movement is not significant between two continuous frames, so the region of interest (ROI) at time $t$ is determined according to the rectangle at time $t-1$. If the positions of vehicles are higher in the image, the movements will become smaller. As a result, the extension is dynamically determined in Fig. 5, where $(X_{\mathrm{TL}}, Y_{\mathrm{TL}})$ and $(X_{\mathrm{BR}}, Y_{\mathrm{BR}})$ are the corresponding top-left and bottom-right positions of the rectangle in the previous frame. $(X'_{\mathrm{TL}}, Y'_{\mathrm{TL}})$ and $(X'_{\mathrm{BR}}, Y'_{\mathrm{BR}})$ are the positions of the extended region. The extension size $\lambda$ has a value that ranges from one block size to three block sizes, depending on the position of the vehicle. The search is performed by vertically scanning on the moving-edge histogram. To reduce noise effect, the edge histogram is smoothly modified using a low-pass filter with a size that is equal to one block size. First, the top position is searched to find the maximum edge density from the central side to the top side in the ROI. Similarly, the bottom boundary is also determined by density scanning. Other than the end-scanning

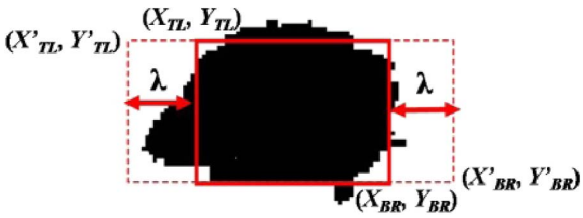



Fig. 6. Diagram of the ROI extension.

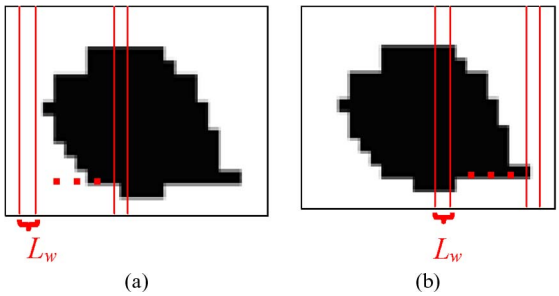



Fig. 7. Diagram of side scanning. (a) Left-side scanning. (b) Right-side scanning.

conditions, the height is determined based on the top and bottom boundaries, which should be less than or equal to the original height. This is because the appearance sizes of vehicles decrease as they leave the detection zone.

Step 2: After obtaining the top and bottom boundaries of the tracked objects, the next step is to search the left and right boundaries. Because of the ambiguous left and right boundaries of moving edges, the left and right boundaries are searched by reference to the foreground image. However, the foreground image needs to be compensated before beginning the search for the boundaries of the right and left sides. Object positions are determined based on the densities of the foreground image, so the search result is sensitive to a broken foreground image. Thus, the broken regions must be filled to achieve better search results. The search of the ROI is further extended in Fig. 6. $(X_{\mathrm{TL}}, Y_{\mathrm{TL}})$, and $(X_{\mathrm{BR}}, Y_{\mathrm{BR}})$ are the corresponding top-left and bottom-right positions of the region in the previous frame. $(X'_{\mathrm{TL}}, Y'_{\mathrm{TL}})$ and $(X'_{\mathrm{BR}}, Y'_{\mathrm{BR}})$ are the positions of the extended region. Left- and right-side scanning from the middle to the two sides determines the density of the rectangle where size $L_w$ is equal to the block size. The scanning-end conditions are also determined based on the density and the boundaries of the ROI rectangle. Left- and right-side scanning are shown in Fig. 7. After applying scanning procedures, the boundaries are determined for the tracked objects.

### B. Verifying the Results of Tracking by FHIIM

To verify the tracking results in the current and previous frame, we used an FHIIM. Two features are utilized in FHIIM, i.e., the color similarity and the area consistency. The color intensity distribution is typically used to assess the tracking results. The proposed method compares the color intensity distribution histograms, because histogram comparison considers all colors in the tracked regions. Higher scores will be obtained if the search results match the previous tracking results. The orders of colors need to be initially reduced, because more colors may be a greater waste of computational resources. In our proposed system, the gray level of each color channel is reduced from 256 to 32. To reduce computational resource consumption, the color similarity measurement $\delta$ is derived using (4), where $H_A$ is the histogram of the search results at time $t$, and $H_B$ is the histogram of the tracking result at time $t-1$. The maximum 1 occurs when the two distributions are the same, whereas the minimum 0 occurs when the two distributions are different.

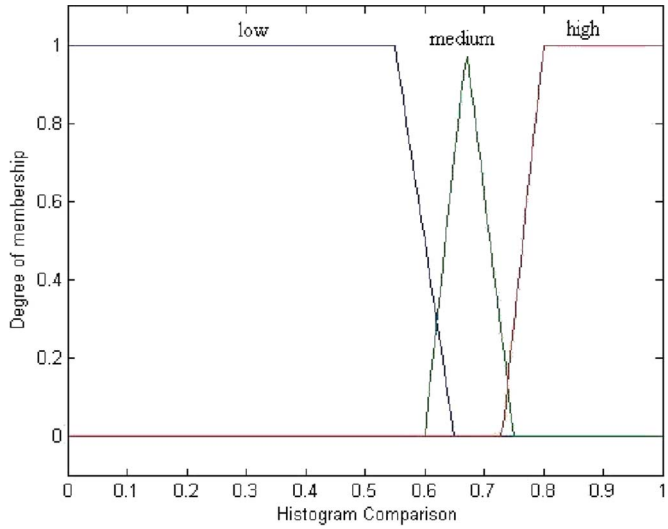



Fig. 8. Membership functions of color comparisons.

$$\delta = \frac{\sum_{i=0}^{32767} Min\left(\hat{H}_A(i), \hat{H}_B(i)\right)}{\sum_{i=0}^{32767} Max\left(\hat{H}_A(i), \hat{H}_B(i)\right)},\ 0 \le \delta \le 1. \tag{4}$$

The movements of the objects must be continuous if the frame rate is sufficiently high. Thus, the correct tracking result must contain the relationships for time $t$, $t-1$, and $t-2$. The search and tracking regions will overlap if the search is successful. $\mathrm{A}_1$ is assumed to be the searched region at time $t$. $\mathrm{A}_2$ and $\mathrm{A}_3$ are tracking regions at time $t-1$ and $t-2$, respectively. In the following, the overlap ratio $R$ is the ratio of the search and tracking results:

$$R = \frac{A_{A_1 \cap A_3}}{A_{A_1 \cap A_2}}. \tag{5}$$

Although information is obtained from the color comparison and area comparison, any direct comparisons are not reliable. Therefore, fuzzy theory, which is denoted as FHIIM, is used to make the comparison more reliable. Three linguistic variables, i.e., "low," "medium," and "high," are defined as the degrees of the color and the area comparisons, which were used as the inputs of the fuzzy inference system. The membership functions of these two inputs are shown in Figs. 8 and 9. The fuzzy logic output variable contains five membership functions, and these five membership functions represent the correctness of tracking. The five membership functions in the output and their linguistic variables are defined as "different," "unlike," "normal," "like," and "same." The membership functions of the output are shown in Fig. 10. The fuzzy logic inference engine is responsible for decision making within the conceptual framework of fuzzy logic and approximate reasoning. The rule base of the relationships between the inputs and outputs is an important part of the fuzzy logic system. Because there are three membership functions in each input variable, there are nine state evaluation fuzzy rules in the rule base of the fuzzy-based behavior decision system, which are shown in Table I.

The output degree is formulated using the fuzzy sets and fuzzy rules. Thus, the coordination of different reactive behaviors can be readily performed using fuzzy reasoning. The process of defuzzification

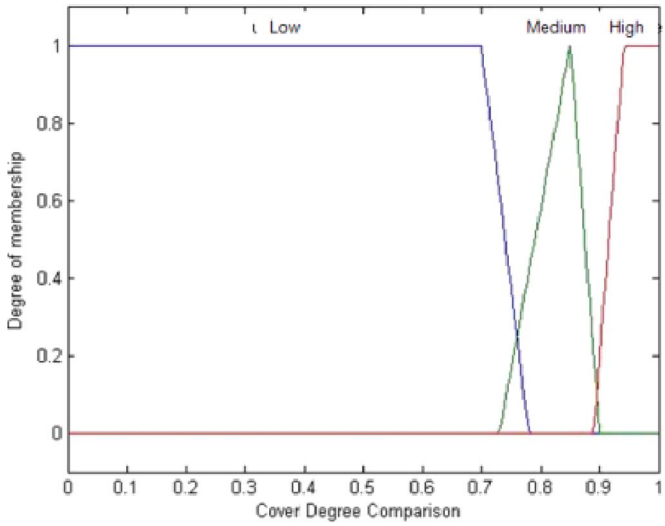



Fig. 9. Membership functions of position comparisons.

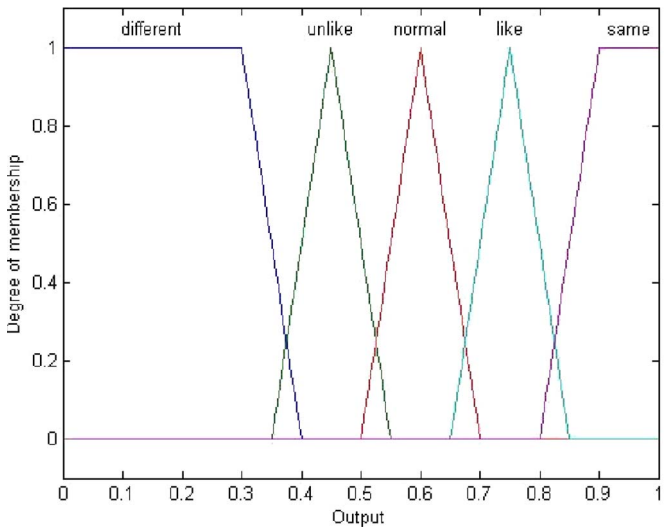



Fig. 10. Output membership function.

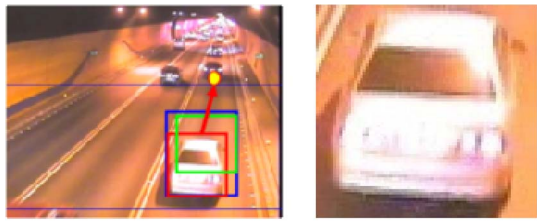
TABLE I
RULE TABLE FOR FUZZY INFERENCE

| δ \ R | High | Medium | Low |
|---|---|---|---|
| High | Same | Like | Unlike |
| Medium | Like | Normal | Unlike |
| Low | Unlike | Unlike | Different |

converts fuzzy information into a crisp result, and the aim of the defuzzification strategy is to make the crisp result perfectly represent the possibility distribution of the inferred fuzzy output. One of the defuzzification methods, i.e., the center-of-area (COA) method provided in

$$Z^*_{\mathrm{COA}} = \frac{\sum_{i=1}^{n} \mu_C(z_i) \times z_i}{\sum_{i=1}^{n} \mu_C(z_i)} \tag{6}$$

where $z_i$ represents the amount of the output at the quantization level $i$, and $\mu_C(z_i)$ represents its membership value, is used here.



(a) (b)

Fig. 11. Diagram showing the prediction for error compensation. (a) Updating to the position with the minimal SAD value. (b) Original pattern.

### C. Tracking Compensation and Masking the Tracked Vehicles

In general, the tracking procedure does not easily lose its target under noncongested conditions. Although FHIIM is used, some unpredictable conditions still occur, where the inference results may differ under congested conditions. We propose a mechanism to prevent these situations, which compensates for this uncertainty. First, the ROI with a blue rectangle is extended, as shown in Fig. 11. Then, the SAD for each position in the ROI is computed to determine the difference from the original pattern at time $t-1$. Tracking is then updated for the position where the SAD value is minimal. Finally, the mask is used to eliminate the tracked objects from the foreground image when the tracked rectangle is determined. If the tracked objects are eliminated from the foreground image, the untracked objects will be correctly labeled by connected component labeling.

## VI. EXPERIMENTAL EVALUATION

In this section, the performance of the proposed system was evaluated using traffic surveillance videos. This analysis and the comparison are discussed in detail. Two scenarios were tested in the experimental evaluation, i.e., Chien-Kuo Bridge in Taipei City in the daytime and nighttime using camera 19 and Mu-Cha Tunnel on Highway 3 using camera 27. There were many challenges in both scenarios. The cameras were very old, and the resolution was not very clear in either scenario. In particular, the camera build was not high enough to cope with the tunnel's limitations. As a result, the overlapping of vehicles was a serious problem in the images. These scenarios were all tested using a Windows XP platform with an Intel Core 2 2.14-GHz central processing unit and 2-GB random access memory. The size of image was 320 × 240, the sampling rate of the sequence was 30 frame/s, and the processing time ranged from 17 to 23 ms. Experimental evaluations were conducted in six environmental situations. In the quantitative evaluations of the proposed system, the Recall and Precision given in

$$\text{Recall} = T_p/(T_p + F_n) \tag{7}$$

$$\text{Precision} = T_p/(T_p + F_p) \tag{8}$$

where $T_p$ (true positives), $F_p$ (false positives), and $F_n$ (false negatives) represent the number of correctly detected vehicles, the number of falsely detected vehicles, and the number of missing vehicles, respectively, were used to evaluate the information retrieval performance [18].

### A. Analysis of Congested/Noncongested Situations

The experimental results are shown in Table II.

### B. Analysis of Results Inside and Outside the Tunnel

The Recall and Precision were both worse inside the tunnel under congested conditions than outside the tunnel, because the height of

TABLE II
EXPERIMENTAL RESULTS FOR THE SIX SCENARIOS

| | | | $T_p$ | $F_n$ | $F_p$ | Recall (%) | Precision (%) |
|---|---|---|---|---|---|---|---|
| Congested | Daytime | Left Lane | 978 | 23 | 114 | 97.70 | 89.56 |
| | | Middle Lane | 1035 | 75 | 22 | 93.24 | 97.92 |
| | | Right Lane | 1306 | 24 | 42 | 98.20 | 96.88 |
| | | Total | 3319 | 122 | 178 | **96.45** | **94.91** |
| | Nighttime | Left Lane | 1362 | 37 | 111 | 97.36 | 92.46 |
| | | Middle Lane | 1406 | 47 | 110 | 96.77 | 92.74 |
| | | Right Lane | 1182 | 12 | 125 | 98.99 | 90.44 |
| | | Total | 3949 | 96 | 346 | **97.63** | **91.94** |
| | Tunnel | Left Lane | 1566 | 90 | 147 | 94.57 | 91.42 |
| | | Middle Lane | 1482 | 68 | 137 | 95.61 | 91.54 |
| | | Right Lane | 1288 | 67 | 166 | 95.06 | 88.58 |
| | | Total | 4336 | 225 | 450 | **95.07** | **90.60** |
| Non-congested | Daytime | Left Lane | 460 | 17 | 5 | 96.43 | 98.92 |
| | | Middle Lane | 510 | 3 | 3 | 99.41 | 99.41 |
| | | Right Lane | 411 | 4 | 2 | 99.04 | 99.52 |
| | | Total | 1381 | 24 | 10 | **98.29** | **99.21** |
| | Nighttime | Left Lane | 390 | 31 | 30 | 92.64 | 92.86 |
| | | Middle Lane | 486 | 16 | 50 | 96.81 | 90.67 |
| | | Right Lane | 292 | 14 | 8 | 95.42 | 97.33 |
| | | Total | 1168 | 61 | 88 | **95.04** | **92.99** |
| | Tunnel | Left Lane | 364 | 7 | 35 | 98.11 | 91.23 |
| | | Middle Lane | 471 | 16 | 20 | 96.71 | 95.93 |
| | | Right Lane | 301 | 19 | 24 | 94.06 | 92.62 |
| | | Total | 1136 | 42 | 79 | **96.43** | **93.50** |

the camera was much lower, and occlusion problems were serious. However, under noncongested conditions, although the performance inside the tunnel is poor in the day, there is no significant difference between the performance inside the tunnel and the performance at night.

### C. Analysis of Daytime, Nighttime, and Tunnel Situations

Under noncongested conditions, the performance in the daytime was better than during the nighttime and inside the tunnel, because the image was much clearer, and there was better contrast in the daytime. Therefore, the Recall and Precision were better in the daytime than the nighttime. The Precision in the daytime was also higher than the nighttime and inside the tunnel. Finally, the Recall and Precision were almost the best in the daytime in all scenarios.

### D. Analysis of Three Lanes

Under congestion, the Recall was higher in the right lane than the other two lanes. Under noncongested conditions, the Precision was worse in the left lane than the other lanes, because the vehicle in the left lane was further from the camera. However, the situation was more complex under congested conditions, and there was no obvious disadvantage in the left lane.

### E. Comparisons With Citilog and Method [9]

First, our proposed approach was compared to Citilog, which is an international company focused on traffic surveillance. Media tunnel is their most advanced video-based monitoring and surveillance product, which allows traffic, security, and safety management operators to identify incidents in real time rather than the traditional passive solutions that are used to verify an incident after it has happened. However, it is difficult to obtain result images for analyzing false positives and false negatives; therefore, we assessed the performance based on the balance of deficient counting and additional counting. For 1 h of testing, the error was updated every 5 min, and the average error (AE) was determined as defined using

TABLE III
EXPERIMENTAL RESULTS WHEN COMPARING PROPOSED METHOD WITH CITILOG IN A CONGESTED INSIDE-TUNNEL SCENARIO

| Time (min) | Left | | | Middle | | | Right | | |
|---|---|---|---|---|---|---|---|---|---|
| | E1 | E2 | M | E1 | E2 | M | E1 | E2 | M |
| ~5 | -2 | -60 | 152 | 16 | -21 | 142 | 6 | 15 | 133 |
| ~10 | 1 | -81 | 162 | -2 | -18 | 145 | 9 | 43 | 113 |
| ~15 | -1 | -71 | 160 | 7 | -2 | 136 | 8 | 7 | 140 |
| ~20 | 5 | -37 | 146 | 11 | -47 | 145 | 9 | 21 | 136 |
| ~25 | 21 | 37 | 147 | 10 | -26 | 136 | 19 | 13 | 132 |
| ~30 | 4 | -80 | 155 | 8 | -46 | 147 | 1 | 11 | 120 |
| ~35 | 0 | -58 | 149 | 10 | -33 | 142 | 8 | 13 | 112 |
| ~40 | 10 | -59 | 149 | -2 | -47 | 143 | 13 | 11 | 120 |
| ~45 | 2 | -88 | 145 | 6 | -59 | 147 | 12 | 23 | 109 |
| ~50 | 10 | -46 | 148 | 3 | -5 | 128 | 13 | 45 | 107 |
| ~55 | 3 | -34 | 143 | 2 | -8 | 139 | 1 | 14 | 133 |
| AE | **3.62** | 39.09 | | **4.95** | 19.76 | | **7.45** | 16.66 | |

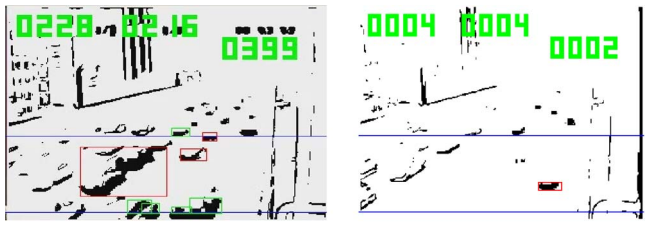
$$\mathrm{AE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{E}{M}\right| \times 100\% \tag{9}$$

where $E$ is the error difference between system counting and manual counting, $M$ is the value for manual counting, and $N$ is the data number. In Table III, $E_1$ is the error with our system, whereas $E_2$ is the error using Citilog. Table III clearly shows that the AE with the proposed approach was less than 8%, whereas the AE with Citilog was greater than 16% under these strict conditions.

The experimental results of Mosabbeb [9] under congested daytime conditions outside the tunnel are listed in Table IV. The Precision was very high, whereas the Recall performance was lower than 77%. Because [9] applied shadows as a feature, this was a faster and simpler approach than the proposed method. In contrast, method [9] had a lower Recall in the outside tunnel scenario because the image was too

TABLE IV
COMPARISONS OF THE RESULTS

| | Left | | Middle | | Right | | Total | |
|---|---|---|---|---|---|---|---|---|
| | [9] | FHIIM | [9] | FHIIM | [9] | FHIIM | [9] | FHIIM |
| Recall | 72.53 | 97.70 | 65.59 | 93.24 | 76.92 | 98.20 | 71.98 | 96.45 |
| Precision | 98.91 | 89.56 | 98.78 | 97.92 | 99.42 | 96.88 | 99.08 | 94.91 |
| Accuracy | 71.95 | **87.71** | 65.06 | **91.43** | 76.57 | **95.19** | 71.51 | **91.71** |



(a) (b)

Fig. 12. Results using method [9]. (a) Connected shadow. (b) Shadow feature ineffective due to high brightness.

(a) (b)

Fig. 13. Comparison of results. (a) Proposed FHIIM. (b) Mosabbeb [9].

bright to detect shadow features and two vehicles were recognized as one vehicle due to their connected shadows, as shown in Fig. 12.

Table IV shows the comparison results. In general, the Recall was higher with the proposed FHIIM than that of [9], whereas the Precision of [9] was higher than that of the proposed FHIIM. This was because the number of detected vehicles (true positives) was much less than FHIIM, resulting in a lower likelihood of falsely detecting vehicles. Therefore, a third performance index was used to make a fair comparison, i.e., the Accuracy using (10). Definitions $T_p$ (true positives), $F_p$ (false positives), and $F_n$ (false negatives) are the same as the definitions previously mentioned. As shown in Table IV, the Recall of FHIIM is greater than 93%, whereas the Accuracy was greater than 87%. However, the Recall of [9] was less than 77%, whereas the Accuracy was less than 77%. Some comparative results are shown in Fig. 13. With method [9], the two vehicles in the left lane and the center lane were recognized as one vehicle because they were visually connected by a shadow

$$\text{Accuracy} = T_P/(T_p + F_p + F_n). \tag{10}$$

## VII. CONCLUSION

We have proposed a new approach for a traffic monitor system that overcame the problems of congested conditions by using FHIIM. Many challenges have been resolved in the experimental scenarios. First, the background has been smoothly updated using SDE, whereas a small range update has been used under congested conditions. Second, block-based segmentation has been used to replace the traditional pixel-based method. Next, the headlights of vehicles may heavily affect the detection ability in the extracted foreground images. Thus, an edge feature method has been used to eliminate the effects of headlights. FHIIM has also been used to compare the final decisions for tracked vehicles. Finally, when tracking errors did appear, error compensation has been used to improve the tracking quality. Based on the experimental results, it has been clear that our proposed approach worked well under normal congested conditions, and it has also provided good performance under congested conditions within a tunnel. Detection within the tunnel has presented many problems. The surveillance equipment was outdated, and the quality of the captured images was very poor. Furthermore, the setting height of the camera was very low, which resulted in bad viewing angles and difficulty in detection. As compared with other methods, the experimental results have shown that the proposed approach had many outstanding performances in terms of detection and tracking.

## REFERENCES


[1] J. Zhang, F. Y. Wang, K. Wang, W. H. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1624–1639, Dec. 2011.

[2] N. Buch, S. A. Velastin, and J. Orwell, "A review of computer vision techniques for the analysis of urban traffic," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 3, pp. 920–939, Sep. 2011.

[3] N. K. Kanhere and S. T. Birchfield, "A taxonomy and analysis of camera calibration methods for traffic monitoring applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 441–452, Jun. 2010.

[4] F. Y. Wang, "Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 630–638, Sep. 2010.

[5] L. W. Tsai, J. W. Hsieh, and K. C. Fan, "Vehicle detection using normalized color and edge map," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 850–864, Mar. 2007.

[6] N. K. Kanhere and S. T. Birchfield, "Real-time incremental segmentation and tracking of vehicles at low camera angles using stable features," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 148–160, Mar. 2008.

[7] Y. Wang, "Joint random field model for all-weather moving vehicle detection," *IEEE Trans. Image Process.*, vol. 19, no. 9, pp. 2491–2501, Sep. 2010.

[8] J. Melo, A. Naftel, A. Bernardino, and J. Santos-Victor, "Detection and classification of highway lanes using vehicle motion trajectories," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 2, pp. 188–200, Jun. 2006.

[9] E. A. Mosabbeb, M. Sadeghi, M. Fathy, and M. Bahekmat, "A low-cost strong shadow-based segmentation approach for vehicle tracking in congested traffic scenes," in *Proc. IEEE Int. Conf. Mach. Vis.*, 2007, pp. 147–152.

[10] W. Zhang, Q. M. J. Wu, G. Wang, and X. You, "Tracking and pairing vehicle headlight in night scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 140–152, Mar. 2012.

[11] Y. L. Chen, B. F. Wu, H. Y. Huang, and C. J. Fan, "A real-time vision system for nighttime vehicle detection and traffic surveillance," *IEEE Trans. Ind. Electron.*, vol. 58, no. 5, pp. 2030–2044, May 2011.

[12] H. Y. Cheng and S. H. Hsu, "Intelligent highway traffic surveillance with self-diagnosis abilities," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1462–1472, Dec. 2011.

[13] S. C. Chen, M. L. Shyu, S. Peeta, and C. Zhang, "Learning-based spatio-temporal vehicle tracking and indexing for transportation multimedia database systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 4, no. 3, pp. 154–167, Sep. 2003.

[14] P. Kumar, S. Ranganath, W. Huang, and K. Sengupta, "Framework for real-time behavior interpretation from traffic video," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 1, pp. 43–53, Mar. 2005.

[15] J. W. Hsieh, S. H. Yu, Y. S. Chen, and W. F. Hu, "Automatic traffic surveillance for vehicle tracking and classification," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 2, pp. 175–187, Jun. 2006.

[16] C. C. C. Pang, W. W. L. Lam, and N. H. C. Yung, "A method for vehicle count in the presence of multiple-vehicle occlusions in traffic images," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 441–459, Sep. 2007.

[17] A. Manzanera and J. C. Richefeu, "A new motion detection algorithm based on $\Sigma - \Delta$ background estimation," *Pattern Recognit. Lett.*, vol. 28, no. 3, pp. 320–328, Feb. 2007.

[18] I. Cohen and G. Medioni, "Detecting and tracking moving objects for video surveillance," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 1999, vol. 1, pp. 319–325.