An Automatic Traffic Surveillance System for Vehicle Tracking and Classification

Shih-Hao Yu, Jun-Wei Hsieh*, Yung-Sheng Chen, and Wen-Fong Hu

Department of Electrical Engineering, Yuan Ze University,
135 Yuan-Tung Road, Chung-Li 320, Taiwan, R.O.C.
{shieh,eeyschen}@saturn.edu.tw

Abstract. This paper presents an automatic traffic surveillance system to estimate important traffic parameters from video sequences using only one camera. Different from traditional methods which classify vehicles into only cars and non-cars, the proposed method has a good capability to categorize cars into more specific classes with a new "linearity" feature. In addition, in order to reduce occlusions of vehicles, an automatic scheme of detecting lane dividing lines is proposed. With the found lane dividing lines, not only occlusions of vehicles can be reduced but also a normalization scheme can be developed for tackling the problems of feature size variations. Once all vehicle features are extracted, an optimal classifier is then designed to robustly categorize vehicles into different classes even though shadows, occlusions, and other noise exist. The designed classifier can collect different evidences from the database and the verified vehicle itself to make better decisions and thus much enhance the robustness and accuracy of classification. Experimental results show that the proposed method is much robust and powerful than other traditional methods.

1 Introduction

Intelligent Transportation System (ITS) is the application which incorporates electronic, computer, and communication technologies into vehicles and roadways for increasing traffic safety, reducing congestion, and thus improving people's life quality. Due to the ease of maintenances and high flexibility in automatic traffic parameter extraction via image processing, there have been a number of different vision-based systems proposed in the literature [1-5]. For example, Beymer *et al.* [1] proposed a vehicle tracking algorithm to estimate traffic parameters based on corner features. Additionally, Gupte *et al.* [2] proposed a region-based approach to track and classify vehicles based on the establishment of correspondences between regions and vehicles. However, in their approaches, only two categories, i.e., cars and non-cars, are handled. Therefore, Sullivan *et al.* [3] proposed a 3D model matching scheme to classify vehicles into various types like wagons, sedan, hatchback, *etc*. Although 3D features obtained from stereo cameras may be useful for categorizing various types of vehicles, the inherent correspondence problem makes them unfeasible for real-time applications.

---

* To whom all correspondence should be addressed.

Except the correspondence problem, several environmental variations will much affect the accuracy of the whole surveillance systems. For examples, shadows will cause two vehicles connect together and lead to the failure of vehicle classification. In addition, perspective distortion will lead to that the vehicle features such as length, width, height are not constant when vehicles move. More importantly, many different vehicle types have similar features, which cause that most approaches can classify vehicles only into two simple categories, i.e., cars and non-cars.

In this paper, we propose a novel vehicle surveillance system to detect, track, and classify vehicles into different classes. First of all, this system uses a technique of image differencing to detect different vehicles from video sequences. Then, a Kalman filter is designed for tracking different vehicles and then obtaining their trajectories. After that, a new defined "linearity" feature is extracted from each vehicle. The new defined "linearity" feature is very useful for discriminating "bus" from "track" without using any 3D information. To reduce the occlusion problem of vehicles, this paper presents an automatic scheme to detect all possible lane dividing lines by analyzing different vehicles' trajectories. Then, the found lane dividing lines can be very effectively used for solving the problems of vehicle occlusions. In practice, due to perspective distortion, the dimension of a vehicle will gradually change when it movements. The found lane dividing lines also can provide important information to scale the vehicle dimension. After feature extraction, an optimal classifier is then designed for accurate vehicle classification. When designing the optimal classifier, we should know that vehicle features are easily affected by shadows or light changes. Since a vehicle has many appearances when it moves, these appearances can provide quite supports for designing the desired classifier. Based on this idea and the spirit of maximum likelihood estimation, an optimization classifier is then designed for robust vehicle categorization. Experimental results show the proposed method offers great improvements in terms of accuracy, robustness, and stability in traffic surveillance.
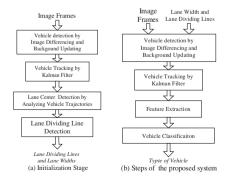
2   Overview of the Proposed System

Fig. 1 Details of the proposed system.

In this paper, we propose a novel traffic surveillance system for estimating traffic parameters from video sequence. This system includes an initialization stage to obtain the information of lane widths and lane dividing lines. Then, traffic parameters can be estimated through image differencing, Kalman filter, feature extraction, and classification by an optimal classifier. In what follows, details of each procedure are discussed.

3   Vehicle Segmentation, Tracking, and Lane-Dividing Detection

In order to simplify the problems of vehicle segmentation, this paper assumes the analyzed image frames are captured by a still camera.

3.1   Vehicle Segmentation and Tracking

When the camera is static, the moving objects can be detected through background subtraction. Assume I_k and B_k are intensities of the kth frame and the kth background, respectively. The difference image $D_k(x, y)$ can be defined as follows:

$$D_k(x, y) = \begin{cases} 0, & \text{if } |I_k(x, y) - B_k(x, y)| \le T_d; \\ 1, & \text{otherwise}, \end{cases} \qquad (1)$$

where T_d is the average value of $|I_k(x, y) - B_k(x, y)|$. After thresholding, each moving vehicle can be segmented from input video frames. Then, in order to obtain the trajectory of each vehicle, a Kalman filter [6] is designed for tracking all possible locations of the detected vehicle. Due to the limited space of this paper, details of designing the used Kalman filter are not discussed here.

3.2 Lane-Dividing Detection

In practice, due to shadows, different vehicles will often connect together. Such occlusions will disturb the accuracies of traffic parameter estimation. In what follows, we will present an automatic scheme to detect possible lane dividing lines from videos for overcoming the occlusion problem of vehicles. Let O_k^i denote the ith moving vehicle at the kth frame. In general, if the vehicle O_k^i moves regularly, the center of O_k^i should be very close to the center of its corresponding lane.   According to this idea, a lane-dividing line detection algorithm can be descried as follows.

Lane-Dividing Line Detection Algorithm

S1: Create a matrix $H_{vehicle}(x, y)$ and initialize all the entries of $H_{vehicle}$ to be zero.

S2: For each vehicle O_k^i, do $H_{vehicle}(x_{O_k^i}, y_{O_k^i}) + = 1$.

S3: Let $A_j = \sum_i H_{vehicle}(i, j) / N_{col}$. Detect all the peaks of $H_{vehicle}$ along the jth row if its value in $H_{vehicle}$ is larger than A_j. Collect the largest m peak positions $x_{j,k}$ as
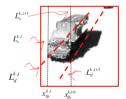
the set $X_j = \{x_{j,k}\}_{k=1,...,m}$.

S4: Let L_C^k be the kth lane central line.   Then, L_C^k can be calculated as: $L_C^k = \{(x_{j,k}, j)^t\}_{j=1,...,N_{row}}$, where $x_{j,k}$ is the kth element of X_j .

S5: With two adjacent L_C^k and L_C^{k+1} , the lane dividing line L_d^k can be detected as: $L_d^k = \{p_k(j)/2 + p_{k+1}(j)/2\}_{j=1,...,N_{row}}$, where $p_m(j)$ is the jth point in C_m .

3.3 Occlusion Detection and Eliminaiton

As described before, different vehicles will often connect together and form occlusions due to shadows. In order to accurately estimate traffic parameters, this paper take advantages of the found lane-dividing lines to separate occluded vehicles into different parts. Like **Fig. 2**, assume $L_d^{k,i}$ is one of lane dividing lines passing through the vehicle V^k . We denote $x_{lb}^{k,i}$ as the x coordinate of the most left-bottom pixel appearing both in $L_d^{k,i}$ and V^k . With $x_{lb}^{k,i}$, we can define a vertical line $L_v^{k,i}$ as:

$$L_v^{k,i} = \{(x, y) \mid x = x_{lb}^{k,i}\} \ . \tag{2}$$

With $L_v^{k,i}$, the vehicle V^k can be divided into two different parts, i.e., the left part $V_{L_v^{k,i}}^{left}$ and the right part $V_{L_v^{k,i}}^{right}$. In addition, with $L_d^{k,i}$, V^k can be divided into the left part $V_{L_d^{k,i}}^{left}$ and the right part $V_{L_d^{k,i}}^{right}$. Clearly, if V^k has no occlusion with other vehicles, the size of $V_{L_v^{k,i}}^{left}$ should be quite different to $V_{L_v^{k,i}}^{right}$. Otherwise, one of $V_{L_d^{k,i}}^{left}$ and $V_{L_d^{k,i}}^{right}$ should be a shadow region. According to these two conditions, a vehicle whether it has occlusions with other vehicles can be detected.
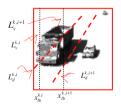

Fig. 2 Different cases of vehicles. (a) Vehicle with no occlusion. (b) Vehicle with occlusion.

If occlusions happen, the vehicles should be first separated into different parts before classification. In what follows, we will propose a novel algorithm to eliminate vehicle occlusions as can as possible. Here, assume V^k is an occluded vehicle and $L_d^{k,i}$ is the dividing line passing V^k .

Vehicle Occlusion Elimination Algorithm

S1: With $L_d^{k,i}$, divide V^k into two different parts: $\overline{V}_{L_d^{k,i}}^{left}$ and $\overline{V}_{L_d^{k,i}}^{right}$.

S2: Let $C_{k,i}^{left}$ and $C_{k,i}^{right}$ be the centrals of $\overline{V}_{L_d^{k,i}}^{left}$ and $\overline{V}_{L_d^{k,i}}^{right}$, respectively. Determine a

straight line $L_C^{k,i}$ passing through $C_{k,i}^{left}$ and $C_{k,i}^{right}$.

S3: For each p_j in $L_C^{k,i}$, decide a vertical line $L_{vertical}^{k,i,j}$ with the equation $x = x_{p_i}$ and a

line $L_{parallel}^{k,i,j}$ with the form $y = m_i^k x + (m_i^k x_{p_i} - y_{p_i})$.

S4: Let $S_{verical}^{k,i,j}$ be the set of pixels which appear both in $L_{verical}^{k,i,j}$ and V^k . Then, we can

get an index j_v as: $j_v = \arg\min_j |S_{vertical}^{k,i,j}|$.

S5: Let $S_{parallel}^{k,i,j}$ be the set of pixels which appear both in $L_{parallel}^{k,i,j}$ and V^k . Then, we

can get an index j_p as: $j_p = \arg\min_j |S_{parallel}^{k,i,j}|$.

S6: Obtain the separation line $L_{separate}^{k,i}$ by: $L_{separate}^{k,i} = \begin{cases} L_{vertical}^{k,i,j_v}, & \text{if } |S_{parallel}^{k,i,j_v}| > |S_{verical}^{k,i,j_p}|, \\ L_{parallel}^{k,i,j_p}, & \text{otherwise.} \end{cases}$

S7: Separate V^k into different parts with the rules:

$V_l^{k,i} = \{p \mid p \in V_k \text{ and } L_{separate}^{k,i}(p) < 0\}$ and $V_r^{k,i} = \{p \mid p \in V_k \text{ and } L_{separate}^{k,i}(p) > 0\}$.

4   Vehicle Classification

Once all the input vehicles have been extracted, we should classify them into different categories for traffic parameter estimation. In this section, the new feature "linearity" of vehicles will be defined and extracted for effective vehicle classification. In what follows, details of feature extraction and classifier designing are discussed.

4.1 Feature Extraction

As described before, this paper uses "size" and "linearity" features to classify vehicles into categories. For the first feature, due to the perspective effect, the size of a vehicle will gradually change when it moves. Therefore, before classification, the size feature should be normalized in advance. Assume $\{L_d^i\}$ is a set of lane dividing lines found in Section 3.2. With the set $\{L_d^i\}$, the width of the ith lane L_d^i at the jth position can be calculated by: $W_{Lane_i}(j) = |p_{L_d^i}^x(j) - p_{L_d^{i+1}}^x(j)|$, where $p_{L_d^i}(j)$ is the jth point in L_d^i . Assume v_h^i is one vehicle appearing in the ith lane, $y_{v_h^i}$ and $S_{v_h^i}$ are the y coordinate and size of v_h^i , respectively. Then, $S_{v_h^i}$ can be then normalized as follows:

$$\overline{S}_{v_h^i} = S_{v_h^i} / W_{Lane_i}^2(y_{v_h^i}). \tag{3}$$

Through normalization, $\overline{S}_{v_h^i}$ forms a good feature for vehicle classification. In addition to vehicle size, the "linearity" feature is also very important for vehicle categorization. Like **Fig. 3**, the bus and truck have similar sizes and speeds but the up-slanted edge of a bus is quite different to that of a truck. If the "linearity" of an up-slanted edge can be defined, classifying vehicles into buses or trucks will become easy. Let

U_{H_i} be the set of up-slanted edge pixels of a vehicle H_i. Assume U_{H_i} has N data points (x_i, y_i). It can be fitted to a straight-line model: $y = mx + b$. The parameters m and b can be easily obtained by minimizing $\sum_{i=1}^{N}(y_i - b - mx_i)^2$ with a numerical method. Then, the linearity of H_i is defined as:

$$Linearity(H) = \exp(-\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - mx_i - b)^2}).$$

Fig. 3 Different up-slanted boundaries of a bus and a truck.

4.2 Classification by Vehicle Library

In this section, an optimal classifier will be designed for categorizing vehicles into different classes. In the past, most classification schemes categorized vehicles based on only one vehicle sample. If this sample is corrupted by noise, they will often fail to work. In fact, we can observe that a vehicle has many different appearances when it moves alone one lane. If the designed classifier can integrate more cues from these appearances and evidences from the database, much robustness and accuracy can be achieved when classifying vehicles. Assume there are K classes in the database and the kth class VC_k has n_k samples. Let V_j^k be the jth vehicle in VC_k and $f_r(V_j^k)$ the rth feature of V_j^k. Then, given a vehicle H_i and a sample V_j^k in VC_k, the similarity between H_i and V_j^k can be defined as:

$$S_k(H_i, V_j^k) = \exp\left(-\sum_{r=1}^{2}(f_s(H_i) - f_r(V_j^k))^2 / \sigma_{r,k}^2\right), \tag{4}$$

where $\sigma_{r,k}$ is the variance of $f_r(V_j^k)$. Then, given a class VC_k, the similarity of H_i to VC_k can be defined as:

$$S(H_i | VC_k) = \frac{1}{n_k}\sum_{V_j^k \in VC_k} S_k(H_i, V_j^k).$$

Furthermore, the probability of H_i belonging to VC_k can be calculated by:

$$P(VC_k | H_i) = S(H_i | VC_k) / S_{sum}(H_i), \tag{5}$$

where $S_{sum}(H_i) = \sum_{k} S(H_i | VC_k)$. Let H_i^t be the detected result of H_i at the tth time

frame. All the H_i^t will be integrated together for improving the accuracy of vehicle classification. Then, for each H_i, the system classifies it into the class l when

$$\forall k \neq l, \quad \sum_{H_i \in Q_{H_i}} P(VC_l | H_i) \geq \sum_{H_i \in Q_{H_i}} P(VC_k | H_i), \tag{6}$$

where Q_{H_i} is the set of all appearances H_i^t.

5 Experimental Results

In order to analyze the performance of the proposed method, a series of image sequences were used. **Fig. 4**(a) shows the detection results of dividing line with two thousands of vehicles. **Fig. 4**(b) shows the results when tracking different occluded vehicles. These occluded vehicles are difficultly separated if the information of dividing lines is not used. However, they can be well separated, detected, and tracked with our proposed method. **Fig. 5** (a) shows the results of vehicle classification. The trajectory of each vehicle is shown by a tail curve line and the plus symbol denotes the vehicle is classified as a car. **Fig. 5** (b) shows other classification results when various vehicles appear in the video frames. Here, the symbols "∇" and "∆" are used to denote a "truck" and a "bus", respectively. If only the vehicle size is used for classification, it is difficult to classify a bus and a truck into different classes.
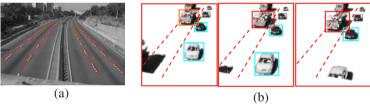

        (a)   (b)

Fig. 4 (a) Results of lane dividing line detection. (b) Results of vehicle tracking when two vehicles connect together due to shadows.

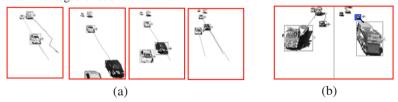   (a)   (b)

Fig. 5 (a) Results of tracking and classification at different frames. The vehicle trajectories are denoted as tail curves. (b) Results of vehicle classification when various vehicles appear.

On the other hand, when classifying, if only one frame is used, quite classification errors will be produced. For example, in **Fig. 6**, the truck is correctly classified in (a), (c), and (d) but misclassified in (b). If the correct results in (a), (c), and (d) can be integrated, the error in (b) should be well corrected. Table 1 lists the classification results when only one frame is used. Table 2 lists the same classification analysis but all the vehicles in the same trajectory are integrated for classification. The superiority

of the proposed method can be verified through the preceding experimental results.
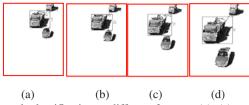

(a)   (b)   (c)   (d)

Fig. 6 Results of truck classification at different frames. (a), (c), and (d) are correctly classi-fied but (b) is misclassified. (b) can be corrected if (a), (c), and (d) are integrated.

Counts / Sequence	vehicles in reality			Correct Classification			Accuracy
	Car	Bus	Truck	Car	Bus	Truck	
Sequence 1	255	62	50	217	53	50	85%
Sequence 2	252	56	63	223	50	61	90%
Sequence 3	247	73	62	203	60	54	83%

Table 1: Different classification accuracies when only one frame is used.

Counts / Sequence	Vehicles in reality	Correct Classification	Success rate of track-ing
Sequence 1	377	344	91%
Sequence 2	371	360	97%
Sequence 3	382	340	89%

Table 2: Different classification accuracies when more frames are used.

References

1. D. Beymer, *et al.*, "A real-time computer vision system for measure traffic pa-rameters," in *Pro. IEEE Conf. CVPR*, Puerto Rico, pp. 496-501, June 1997.
2. S. Gupte, *et al.*, "Detection and classification of vehicles," *IEEE Transactions on ITS*, vol. 3, no. 1, pp. 37-47, March 2002.
3. G. D. Sullivan, *et al.*, "Model-based vehicle detection and classification using orthographic approximations," *Image Vision Computing*, vol. 15, no. 8, pp.649-654, Aug. 1997.
4. Y.-K. Jung, K.-W. Lee, and Y.-S. Ho, "Content-based event retrieval using se-mantic scene interpretation for automated traffic surveillance," *IEEE Transactions on ITS*, vol. 2, no. 3, pp.151-163, Sept. 2001.
5. A. J. Lipton, *et al.*, "Moving target classification and tracking from real-time video," in *Proc. IEEE Workshop Applications of Computer Vision*, 1998, pp.8-14.
6. R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME-Journal of Basic Engineering*, vol. 82, pp.35-45, March 1960.