PAPER *Special Issue on Information System Technologies for ITS*

A Highway Surveillance System Using an HMM-Based Segmentation Method

Jien KATO[†], **Toyohide WATANABE**[†], ***and*** **Hiroyuki HASE**[††], ***Regular Members***

SUMMARY Automatic traffic surveillance based on visual tracking techniques has been desired for many years. This paper proposes a basic highway surveillance system using an HMM-based segmentation method. The presented system meets the essential requirement of ITS: real-time running. Its another advantage is robustness to the shadows of moving objects, which have been recognized as one of main obstacles to robust car tracking. At present, using the system we can estimate velocity of vehicles with high accuracy. For acquiring metric information in the real world, the system does not require a precise calibration but only needs four point correspondences between the image plane and ground plane.

key words: *traffic surveillance, car tracking, hidden Markov model, car speed estimation*

1. Introduction

Utilizing visual information is the most suitable way to grasp circumstances in many cases. From this point of view, automatic traffic surveillance based on computer vision techniques has been of interest for many years [17]. Initial purposes of such surveillance systems include obtaining information about road usage that helps road engineers to find out the areas in need of alteration of existing traffic patterns. Recently, more interest has focused on acquiring information for ITS (Intelligent Transport System) [22]. One required function of the systems is to alert drivers to problems in advance, for example, heavy congestion or other exceptional events ahead on the road. Naturally, such systems need to work in *real time* and to operate robustly in *real-world* traffic environment. In this paper, we propose a basic highway surveillance system based on visual tracking techniques. This system achieves real-time working and is robust to the shadows of moving objects, which have been recognized as one of main obstacles to robust car tracking [9].

Ideally, a highway surveillance system with the objective of providing ITS with a variety of traffic information should perform the following tasks:

- Count vehicles and estimate the speed of vehicles
- Monitor lane occupancy and detect lane-changing
- Classify vehicles into different groups: trucks, cars, cars with trailers, motor bikes, etc.
- Report exceptional events such as congestion, accidents and breakdowns of vehicles

In order to acquire any kind of above information from images, it is necessary to track vehicles reliably over time on traffic surveillance movies. Tracking vehicles have been achieved in different approaches [1], [3], [6], [8], [13]–[15]. Now we survey these approaches according to the dimensionality of the representation for target objects.

Three-dimensional model-based approach usually represents each type of vehicle by the skeleton of the object or a wire frame model. Koller et al. [14], [15] and Baker et al. [1] demonstrated that trajectories of vehicles could be recovered with high accuracy in 3-D-based approach, although the systems were computationally expensive and did not run in real time. The most serious weakness of this approach is the reliance on very detailed models for all vehicles that can be found on roadways.

On the other hand, the idea of region-based tracking is to model each vehicle as a connected region, to identify it in the image and then to track it over time using techniques such as cross-correlation measure [13]. This approach can be evaluated as a two-dimensional model-based one. A common problem with region-based methods is the difficulty in segmenting individual vehicles under congested traffic condition: so-called occlusion problem [8].

Active contour or snake [3] can be regarded as a 1-D model-based approach. The basic idea of this approach is to generate a representation of the bounding contour of an object and to keep dynamically updating it. The merit of having a contour based representation is of course reduction in computational complexity. Ferrier et al. [6] built an active contour-based car tracker that ran in real time.

Distinct completely from the approaches mentioned thus far, feature-based approach abandons the idea of tracking objects as a whole, but instead tracks distinguishable points such as corners of objects. We thus evaluate it as zero-dimensional model-based approach. The advantage of this approach is that even in the presence of partial occlusion, some of the subfeatures of moving objects remain visible. Since ve-

Manuscript received February 28, 2002.
Manuscript revised May 24, 2002.

[†] The authors are with the Department of Information Engineering, Graduate School of Engineering, Nagoya University, Nagoya-shi, 464–8603 Japan.

[††] The author is with the Faculty of Engineering, Toyama University, Toyama-shi, 930–8555 Japan.

hicles have similar features, the difficulty is to find a method to group the feature points. Grouping process should distinguish a vehicle from its neighbors by using motion parameters such as speed, acceleration and lane drift. The grouping algorithms are usually complex and computationally expensive.

In effect, acquiring most of desired traffic information (as listed above) does not require precise 3-D modeling of the objects. From this viewpoint, the highway surveillance system presented in this paper exploits an HMM-based (hidden Markov model) segmentation method as a low-level car tracker, which essentially belongs to the region-based approach. This method has two characteristic aspects. On one hand, it deals with the problem of shadows of moving objects. Shadows of vehicles obstruct robust car tracking, however, it is difficult to remove the shadows by using traditional image differentiation techniques (either background subtraction or inter-frame differentiation). As yet we have not found any existing method which is able to track vehicles excluding the shadows. Our method alleviate this problem considerably by modeling shadows as well as foreground and background objects through one HMM. The basic idea of the segmentation method has been addressed in [9], [10]. On the other hand, although image segmentation based on HMMs is generally computationally expensive, in our system a mechanism realized by a forward procedure of state estimation makes real-time car tracking possible with modest hardware. In Sect. 2, we first give the approach of our highway surveillance system. Section 3 summaries the HMM-based segmentation method for completeness, and then Sect. 4 extends this method into a car tracker via introducing context-dependence among neighboring HMM regions. In Sect. 5, we set out a simple calibration for real-time recovery of Cartesian trajectories of vehicles based on only four point correspondences between the image and ground, rather than a precise calibration as described in [5]. Experimental results of estimating velocity of vehicles on real-world video sequences are shown and discussed in Sect. 6. Finally, we draw conclusions and point out future work in Sect. 7.

2. Approach

In our approach, tracking vehicles means to perform accurate segmentation over time of the foreground objects, vehicles, from background objects and the *shadows* (of vehicles). Shadows of moving objects are regarded as an individual category because they move but do not belong to the foreground. Previous work [12], [21] has revealed that such shadows are one of the main factors that disturb tracking process and need to pay special attention. We employ an hidden Markov model to classify and segment each field image of traffic movies into three different categories: foreground (**F**), background (**B**) and shadow (**S**). The motivation of using HMMs springs from two aspects. Firstly, an HMM is suitable to incorporate *temporal continuity*, namely, the nature that a pixel belongs to a certain category for a period of time. The use of temporal continuity enhances the performance of segmentation of the system. Secondly, HMMs allow to learn model parameters from an ordinary image sequence. Being free of requirement of specific learning data leads to considerable simplification in model parameter training stage.

Images of a surveillance movie are divided into non-overlapping small regions with equal size (called HMM region). Each *region location* is modeled as an HMM (1-D HMM) to take into account the temporal continuity of different categories along a *time-axis*[†]. Intensity and wavelet coefficients in high frequency bands are used as observations in our approach. The use of the second observation is based on the idea that the variance of wavelet coefficients in high frequency bands should be small for **S** and **B**, but large for **F**, since the foreground objects are usually sharply focused and have more details within the objects than the out-of-focus background and shadow regions.

The system is mainly composed of two phases: the learning phase and the tracking phase. In the learning phase, the system learns the unknown HMM parameters with an EM-type (Expectation-Maximization) algorithm [4] for individual region locations or relearns the parameters when necessary, based on the observations over several seconds of a video sequence. In the tracking phase, the system classifies each region in a field image of a movie into **F**, **B** and **S** by a state estimation algorithm, given the learned model and observed data. This state estimation algorithm performs *context-dependent* (or spatial) classification of individual regions in each field image. As a result, the foreground regions can be segmented out from background and shadow regions over time.

3. HMM-Based Segmentation Method

3.1 The Hidden Markov Model

It is postulated that there are three states in the HMM and that there is a unique mapping from the states to the categories. At any discrete unit of time, an HMM region is assumed to exist in one of the states. Transitions between the states take place according to a probability, depending only on the state of the system at the immediately preceding unit of time (a first-order Markov). Two observations (intensity and wavelet coefficients in high frequency bands) are treated as a 2-D feature vector. At each unit of time, a feature vector is generated from the current state according to a probability distribution, depending only on this state. The

[†] All region locations have the same model but different model parameters.

states in the HMM represent abstract quantities (the categories **F**, **B** and **S**). They can never be observed but correspond to the "clusters" of contexts that have similar probability distributions of the observable feature vectors [16].

The model is characterized by the following parameters: the initial state distribution, the state transition probabilities and the observation probability distributions for each category. Let $S = \{S_f,\ S_b,\ S_s\}$ be the states corresponding to categories **F**, **B** and **S**. The parameters of the HMM, notated as $\Omega = \{A, B, \pi\}$, are defined as follows:

- Initial state distribution: $\pi = \{\pi_b, \pi_s, \pi_f\}$, $\pi_i = \Pr(S_i \text{ at } t = 1)$.
- State transition matrix:
$$A = \begin{pmatrix} a_{bb} & a_{bs} & a_{bf} \\ a_{sb} & a_{ss} & a_{sf} \\ a_{fb} & a_{fs} & a_{ff} \end{pmatrix},$$
$a_{ij} = \Pr(S_j \text{ at } t+1 | S_i \text{ at } t)$.
- Observation probability distribution in state j: $B = \{b_j(v)\}$, $b_j(v) = \Pr(v \text{ at } t | S_j \text{ at } t)$, where v is the 2-D feature vector generated by the first and second observations: intensity and wavelet coefficients.

We approximate the probability distributions of **B** and **S** by 2-D Gaussian densities, i.e.

$$b_i(v) = \frac{1}{\sqrt{(2\pi)^2 \det(\Sigma_i)}} e^{-\frac{1}{2}(v-\mu_i)^t \Sigma_i^{-1}(v-\mu_i)} \qquad i \in \{b, s\}. \tag{1}$$

The mean vector is denoted by $\mu_i = (\mu_1^i, \mu_2^i)$ and the covariance matrix by

$$\Sigma_i = \begin{pmatrix} \sigma_{11}^i & \sigma_{12}^i \\ \sigma_{21}^i & \sigma_{22}^i \end{pmatrix}$$

where the subscripts 1 and 2 mean the first and second observations, respectively. On the other hand, we model the probability distribution of **F** by a uniform probability density, namely,

$$b_f(v) = \frac{1}{256 \times 512} \tag{2}$$

where 256 corresponds with the gray level of the first observation and 512 gives a range of variation of the second observation.

3.2 Parameter Learning and Initialization

The aim of parameter learning is to find the model parameter Ω which maximizes $L(V, \Omega) = \log[p(V|\Omega)]$ for a given set V of observed data. We use Baum-Welsh algorithm [2] to generate a sequence of estimates for Ω given V, so that each estimate Ω^i has a greater value of $\log[p(V|\Omega)]$ than the preceding estimate Ω^{i-1}. The re-estimation formulae for π, A and B are defined as

$$\overline{\pi}_i = \gamma_1(i) \tag{3}$$

$$\overline{\mu}_i = \frac{\sum_{t=1}^{T} v_t \gamma_t(i)}{\sum_{t=1}^{T} \gamma_t(i)} \tag{4}$$

$$\overline{\Sigma}_i = \frac{\sum_{t=1}^{T} \gamma_t(i)(v_t - \overline{\mu}_i)(v_t - \overline{\mu}_i)^t}{\sum_{t=1}^{T} \gamma_t(i)} \tag{5}$$

$$\overline{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \tag{6}$$

where $\gamma_t(i) = \Pr(S_i \text{ at } t | V, \Omega)$ and $\xi_t(i,j) = \Pr(S_i \text{ at } t, S_j \text{ at } t+1 | V, \Omega)$ are auxiliary probabilities that can be efficiently calculated by the so-called forward-backward algorithm [19]. $V = \{v_1, \ldots, v_T\}$ is a sequence of observation symbols.

To set initial parameters properly, time constants τ_b, τ_s and τ_f are defined as the typical duration time in which a pixel belongs to **B**, **S** and **F**. Let also λ_b, λ_s and λ_f be the proportions of the time spent in **B**, **S** and **F** with $\lambda_b + \lambda_s + \lambda_f = 1$. The initial parameters for the state transition matrix are chosen as

$$A = \begin{pmatrix} 1 - \frac{1}{\tau_b} & \frac{1}{\tau_b}\Lambda_{sf} & \frac{1}{\tau_b}\Lambda_{fs} \\ \frac{1}{\tau_s}\Lambda_{bf} & 1 - \frac{1}{\tau_s} & \frac{1}{\tau_s}\Lambda_{fb} \\ \frac{1}{\tau_f}\Lambda_{bs} & \frac{1}{\tau_f}\Lambda_{sb} & 1 - \frac{1}{\tau_f} \end{pmatrix}$$
$$\Lambda_{ij} = \lambda_i / (\lambda_i + \lambda_j) \tag{7}$$

and those for the initial probability to be

$$\pi = \{\lambda_b,\ \ \lambda_s,\ \ \lambda_f\}. \tag{8}$$

The initial parameters for state distributions are determined in the following ways. The mean vector μ_b is initially estimated by the modes of intensities and the wavelet coefficients at a given region location, because $\lambda_b \gg \lambda_s$ and $\lambda_b \gg \lambda_f$ generally hold, while the covariance matrix Σ_b is determined empirically. The initial parameters for μ_1^s and σ_{11}^s are selected based on the assumption that the intensity of the shadow is lower than that of the background, i.e.

$$\mu_1^s = \frac{\mu_1^b + 2\sqrt{\sigma_{11}^b}}{2}, \quad \sigma_{11}^s = \left(\frac{\mu_1^s}{2}\right)^2. \tag{9}$$

The rest initial parameters for μ_2^s and other elements of Σ_s are given the same values as the corresponding parameters for the background.

The model parameters are re-estimated according to Eqs. (3)–(6) until the likelihood probability stabilizes (most around 10 times). Learned probability distributions for **F**, **B** and **S** at one region location are shown in Fig. 1.

4. State Estimation with Spatial Information

The method described in the preceding section models

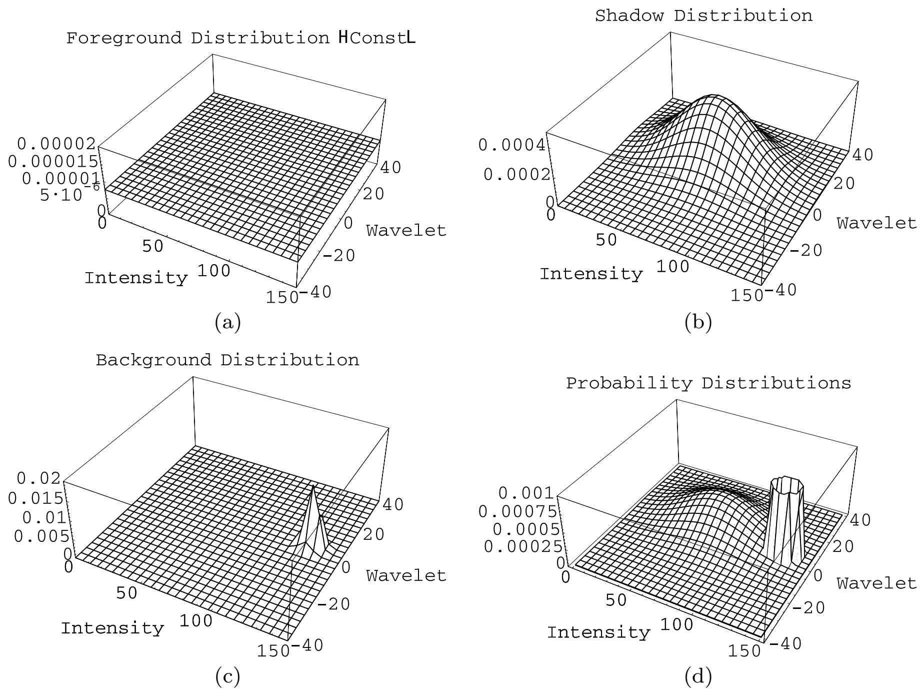


Fig. 1 Probability distributions are learned for S_b, S_s and S_f at one region location. (a) Background, (b) Shadow, (c) Foreground, and (d) The three put together. In (a)–(d), the first observation, intensity, uses output of a 4×4 mean filter, and the second observation, wavelet coefficient, is calculated as the variance of the wavelet coefficients in high frequency bands.

each region location of image sequences *independently* of neighboring region locations along a time-axis. From the viewpoint of tracking, however, it is vital to segment foreground objects out as a *connected* region at each time unit, and moreover to achieve it in real time. We attack this problem by introducing context-dependence among neighboring HMM regions into the HMM-based segmentation method. Concretely, we incorporate spatial information into the criterion for choosing an optimal state sequence associate with a given series of observations. Let the value of a state at region (i,j) be notated by $\mathcal{S}_{i,j}$, and the values of a state set at neighborhood $\mathcal{N}_{i,j}$ of region (i,j) by $\mathcal{S}_{\mathcal{N}_{i,j}}$. The criterion adopted is defined as

$$\underset{k \in \{b,s,f\}}{\operatorname{argmax}} \{\Pr(v_1, \ldots, v_t, S_k \text{ at } t|\Omega) \cdot \Pr(\mathcal{S}_{i,j} = S_k|\mathcal{S}_{\mathcal{N}_{i,j}})\}. \tag{10}$$

The first term in Eq. (10) takes into account the joint probability of the state at time t and the past observations $\{v_1, \ldots, v_t\}$ given the model Ω, from the viewpoint of the *temporal context*. The second term means the probability of the state being S_k at region (i,j), given the probability of state set $\mathcal{S}_{\mathcal{N}_{i,j}}$ at neighborhood $\mathcal{N}_{i,j}$ of region (i,j). Hence, it incorporates spatial relationships among individual HMM regions, from the viewpoint of the *spatial context*. We define $\Pr(\mathcal{S}_{i,j}|\mathcal{S}_{\mathcal{N}_{i,j}})$ as

$$\Pr(\mathcal{S}_{i,j}|\mathcal{S}_{\mathcal{N}_{i,j}}) \propto \exp(\kappa \vartheta(\mathcal{S}_{i,j}; \mathcal{S}_{\mathcal{N}_{i,j}})) \tag{11}$$

where κ indicates a parameter that measures the strength of the context-dependence among the neighboring HMM regions. The function $\vartheta(\mathcal{S}_{i,j}; \mathcal{S}_{\mathcal{N}_{i,j}})$ is selected as

$$\vartheta(\mathcal{S}_{i,j}; \mathcal{S}_{\mathcal{N}_{i,j}}) = \sum_{(s,r) \in \mathcal{N}^8_{i,j}} \frac{1}{16} I(i,j,s,r) + \sum_{(s',r') \in \mathcal{N}^{16}_{i,j}} \frac{1}{32} I(i,j,s',r') \tag{12}$$

$$I(i,j,s,r) = \begin{cases} 1 & \mathcal{S}_{i,j} = \mathcal{S}_{s,r} \\ 0 & \mathcal{S}_{i,j} \neq \mathcal{S}_{s,r} \end{cases} \tag{13}$$

where $\mathcal{N}^8_{i,j}$ and $\mathcal{N}^{16}_{i,j}$ are the 8-neighbors of region (i,j) with distances 1 and 2, respectively. Note that Eq. (10) can be solved by the forward procedure alone [19]. Since the first term in Eq. (10) is defined inductively, it is possible to perform state estimation (segmentation) with Eq. (10) in real time.

In order to segment a foreground object as a connected region, the state estimation with Eq. (10) should be repeated several times. Experiments show that three re-estimations seem sufficient to obtain good results.

The tracking process based on the segmentation method described so far includes three conditions as depicted in Fig. 2. Condition 2 in Fig. 2 denotes resetting the process to start tracking, condition 3 corresponds to tracking a vehicle, and condition 1 is entered when no vehicle is present in the area to be monitored. Fig-

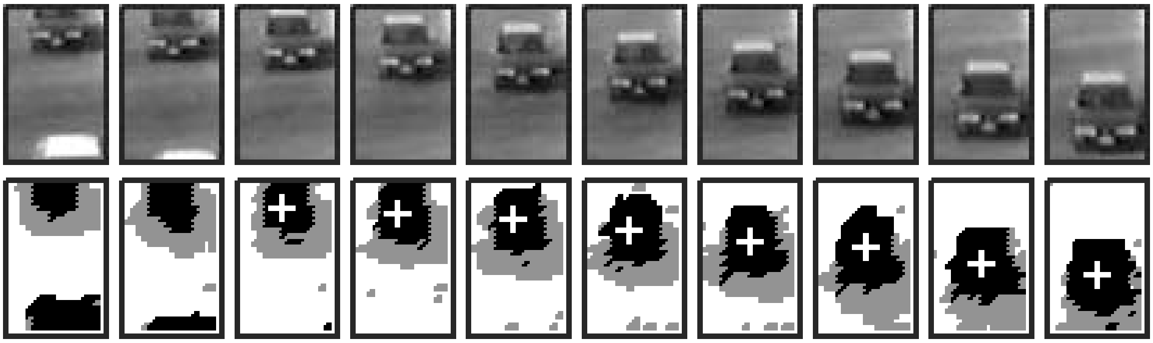


Fig. 3 Tracking a car over an area for testing. In the lower images, foreground, shadow and background are indicated in black, gray and white, respectively. A cross at the center of a foreground region means that the object is being followed by the tracker.

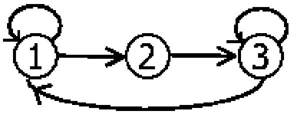


Fig. 2 A three-condition tracking process. Number i indicates condition i.

ure 3 demonstrates that the tracker catches hold of a vehicle when passing through a small area for testing. Multi-vehicles are able to be tracked simultaneously.

5. Acquiring Metric Information in World

This section discusses the real-time recovery of Cartesian trajectories of vehicles when moving on the ground.

Surveillance movies are supposed to be taken from a video camera placed on a bridge or a pole above the traffic area to be monitored, and hence restricted to a certain geometry (Fig. 4). To model the behavior of a perspective mapping between image plane π and ground plane Π, we define a ray model as a set of rays in a 3-D space $\mathcal{R}^3$, in which all rays emanate from a common origin (the center of the lens) and only the direction of a ray is important [18]. A ray intersects the two planes in the corresponding perspectively transformed points, for example, points d and D in Fig. 4. This relationship can be interpreted by considering each of π and Π to be a single plane which is transformed from one plane to the other by transforming the coordinate system of ray space.

It is usually possible to model the arbitrary relationship between π and Π by rotation and anisotropic scaling of the rays in $\mathcal{R}^3$ (affine approximation [18], [20]). Rotation and scaling of $\mathcal{R}^3$ can be represented by a general 3×3 matrix multiplication. Hence, any corresponding points $\mathbf{x}$ and $\mathbf{X}$ on the two planes can be related by a transformation matrix $M(= \{m_{ij}\}; i, j = 1, 2, 3)$ as

$$\lambda \mathbf{x} = M \cdot \mathbf{X} \tag{14}$$

where $\mathbf{x}$ and $\mathbf{X}$ are expressed in homogeneous coordinates: $\mathbf{x} = (x_1, x_2, x_3)^t$ and $\mathbf{X} = (X_1, X_2, X_3)^t$, and λ denotes an arbitrary scalar. The existence of λ means the scale uncertainty. As a result, only eight out of the nine matrix elements are independent and recoverable.

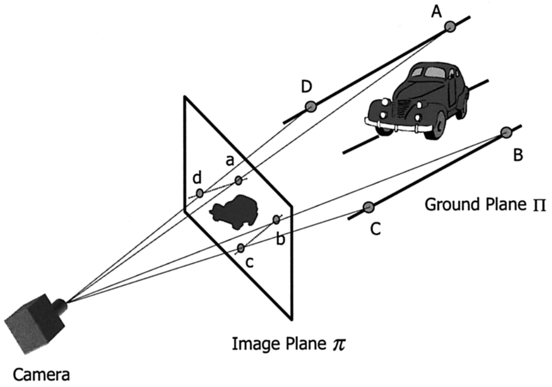


Fig. 4 The geometrical relationship among the camera, the image plane and the ground plane.

If the transformation is represented in Cartesian coordinates, each point in the planes provides two Cartesian coordinate equations, namely,

$$L \cdot M' = H \tag{15}$$

where

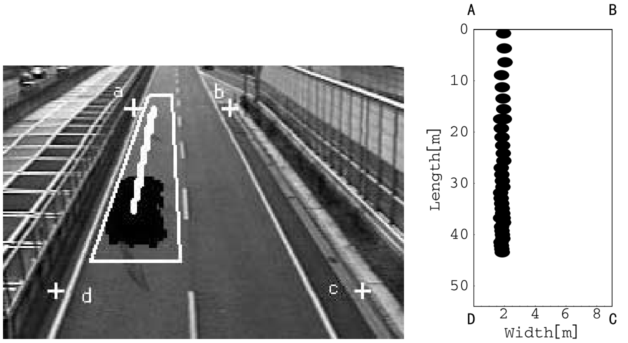
$$L = \begin{pmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_i & Y_i & 1 & 0 & 0 & 0 & -x_iX_i & -x_iY_i \\ 0 & 0 & 0 & X_i & Y_i & 1 & -y_iX_i & -y_iY_i \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix} \tag{16}$$

$$M' = (m_{11}, m_{12}, m_{13}, m_{21}, m_{22}, m_{23}, m_{31}, m_{32})^t \tag{17}$$

and

$$H = (\ldots, x_i, y_i, \ldots)^t. \tag{18}$$

As a result, four point correspondences (no three of those are collinear) between two projectively transformed planes can define the transformation matrix uniquely. For instance, when we have world points A, B, C, D with coordinates $(0,0)$, $(9,0)$, $(9,54)$, $(0,54)$ (unit: meter), and corresponding image points a, b, c, d with coordinates $(0,0)$, $(117,0)$, $(277,221)$, $(-93,221)$ (unit: pixel), the parameters for the projectivity are calculated as $M' = (0.0769231, 0.0323703, 0.0, 1.5546e-17, 0.772711, 0.0, -1.30337e-18, 0.00978458)$.



(a) Camera view of road (b) Plane view of road

Fig. 5 Velocity of vehicles is calculated by the projective transformation between the image and ground planes. World points A, B, C, D and corresponding image points a, b, c, d are used for calibration procedure. The trajectories of the car (moves toward the camera) are represented with filled circles at the positions where the centers of vehicle regions occur. It is worth to note that when a car is farther from the camera, the tracker is less sensitive to the movement of the car. This fact can be taken into account when choosing tracking areas.

6. Experimental Results

Our system is implemented on an SGI O2 R12000 300S entry-level desktop workstation and is able to run at the field-rate of 60 Hz (real time). The tracking process has been tested extensively and the results on real-world highway sequences show that it is possible to accurately distinguish vehicles from background and shadow regions by proposed segmentation method. Some video clips that demonstrate the validity of this method are provided on [11]. In this section, we primarily focus on the experiments of recovering velocity of vehicles, the fundamental traffic information required by a highway surveillance system, using the calibration system described in Sect. 5.

Several driveways around Aichi Prefecture are filmed by a CCD-TR705 Handycam camera (Sony) to acquire testing data for experiments. The reason for selecting driveways rather than highways is to conveniently obtain ground truth information by car speedometers. Surveillance movies are taken from overpasses above traffic areas of interest. This location is subject to vibration from traffic and wind. Each new camera position requires learning of the model parameters and determining the new projective relationship.

The experimental results discussed below are obtained from a video sequence of a time period of one hour. Cars participating in the experiments were required to drive through a specific area, within the camera's visual field, at any nearly constant speed (Fig. 5 (a)). One 30 second video fragment situated in the beginning of the sequence was used to learn the model parameters, and there was no parameter updating during the whole testing period. That means the time intervals between the learning sequence and the testing sequences were up to one hour.

So as to recover the unknown elements in the transformation matrix, M, we choose four point correspondences between the image and the ground planes as shown in Fig. 5. The Cartesian coordinates for these points on the ground plane can be acquired by knowledge about road markers, for instance, the width of roads, the length of line segments in the center of roads and their interval[†]. While, the coordinates of corresponding points on the image plane can be easily obtained from their pixel locations. In the camera position shown in Fig. 5 (a), we select world points A, B, C, D and corresponding image points a, b, c, d. They have the coordinates described in the preceding section, under the identical Cartesian coordinate system. By further setting $m_{33} = 1$ to eliminate the scale λ, the transformation matrix for the projection are hence calculated as follows:

$$M = \begin{pmatrix} 0.0769231 & 0.0323703 & 0.0 \\ 1.5546e-17 & 0.772711 & 0.0 \\ -1.30337e-18 & 0.00978458 & 1.0 \end{pmatrix}. \quad (19)$$

The transformation between the two planes allows us to calculate velocity of vehicles based on frame count and the image displacement of the regions classified as foreground. Figure 5 (a) shows the target car being tracked over an area (trapezoid) consisting of 50 HMM regions in length and each region has 4×4 pixel size. The trajectories of the car on both image and ground are shown in Figs. 5 (a) and (b), respectively, with the centers of detected foreground regions at successive frames. Since the car is first grasped at the 215-th frame in world position $x_w = 1.9365$, $y_w = 0.765224$ ($x_i = 25$, $y_i = 1$ on the image), and released at the 317-th frame in $x_w = 1.85474$, $y_w = 43.4484$ ($x_i = 1$, $y_i = 125$), it moves over 42.753482 meters during 3.4 seconds. That implies a speed of 44.9735 km/h.

Estimated velocities for five appearances of two cars are summarized in Table 1. The second column of Table 1 shows velocities that average every eight frame speeds, and third one give speeds calculated according to the instants a vehicle is grasped and released by a

[†] The area determined by the four points in Fig. 5 (a) has the width same as that of the road, and the length just including three line segments with four intervals on the center of the road.

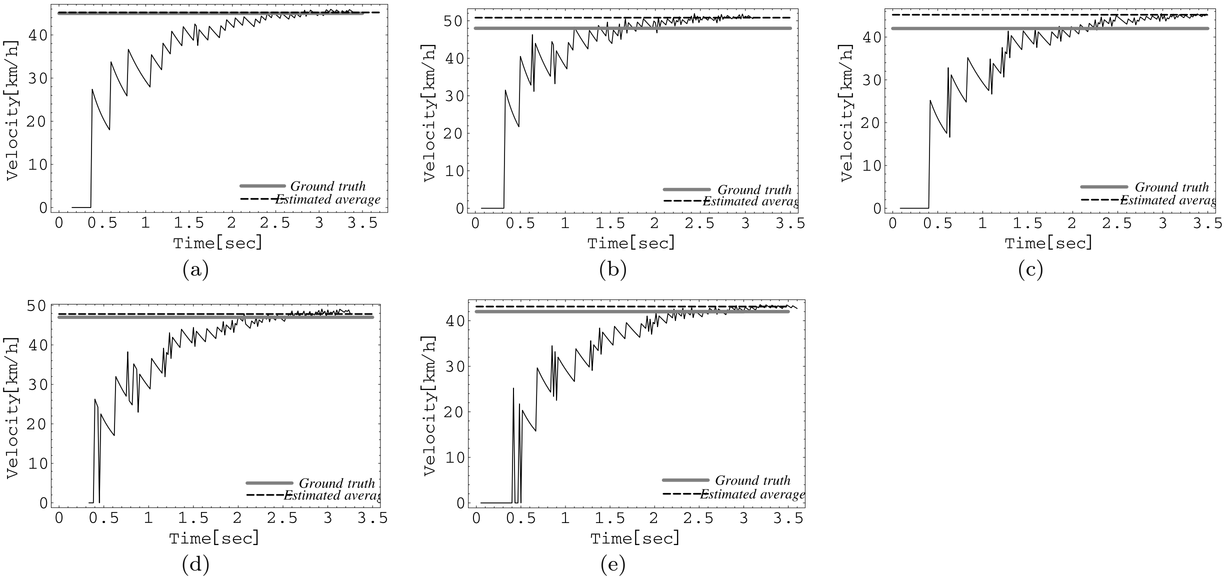


Fig. 6 Estimated velocity of vehicles at each field well converge on the speed given by a car speedometer. (a)–(e) correspond to two cars' five appearances in an area of interest for surveillance. $t = 0$ means the moment the tracker catches the target.

Table 1 Statistics summary.

No.	AV (km/h)	V (km/h)	GT
	Error Rate (%)	Error Rate (%)	(km/h)
1	45.2095	44.9735	45
	0.4566	0.0589	
2	50.83	50.5531	48
	5.8958	5.3190	
3	45.21	45.1455	42
	7.6429	7.4893	
4	47.8271	47.9606	47
	1.7598	2.0438	
5	43.1123	42.6832	42
	2.6483	1.6266	

tracker. Ground truths from car speedometers are indicated in the last column. Error rates between estimated velocities and ground truths are also listed below each approximated value. Many factors, affine approximation of projective transformation, a bigger HMM region size, inaccuracy of the car speedometers, etc. potentially cause the errors, nevertheless the results given in Table 1 offer good precision for car speed estimation.

In addition, we plot estimated velocities at one field interval in graphs (Fig. 6) to provide somewhat intuition how the velocities change as time passes. It can be observed from Fig. 6 that after almost two seconds elapse since the tracker catches a target, the car speed estimated by the system considerably converges on the ground truth. When a car is farther from the camera, the tracker is less sensitive to the movement of the target because the movement is smaller. This is the reason why detected vehicle's centroids in Fig. 5 (b) distribute at larger (precisely irregular) intervals at first. This fact can be taken into account when choosing tracking areas.

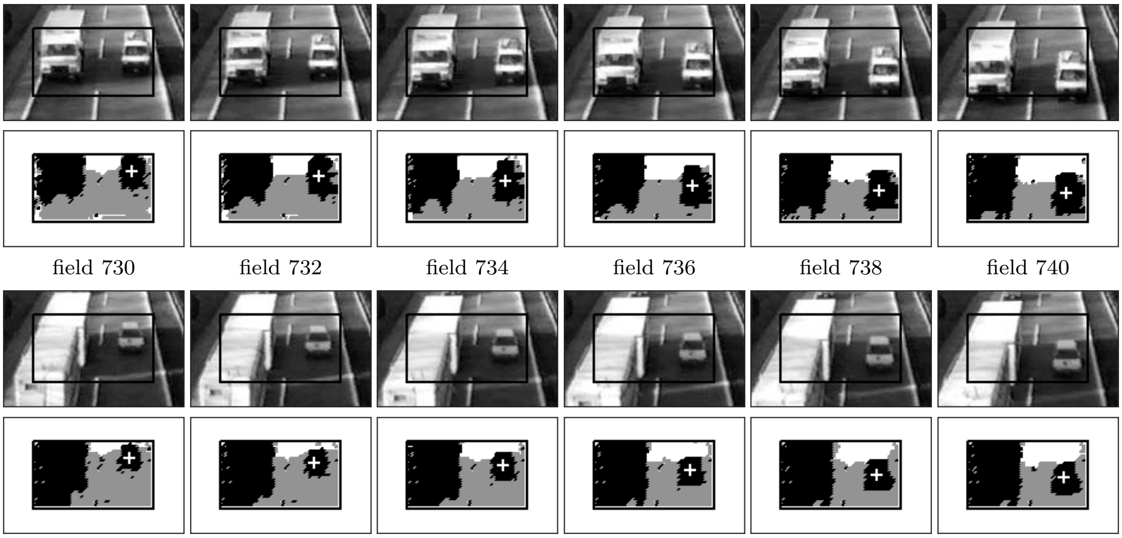
We have emphasized that one characteristic of our method is the capability of tracking vehicles excluding the shadows. This is confirmed by two examples shown in Fig. 7 where the target cars on the right lane, surrounded by large shadows cast by the trucks on the left lane, are reliably tracked over time, independently of the shadows. In the examples both vehicles on left and right lanes are distinguished distinctly, while some traditional methods could detect only the trucks and would miss the vehicles in the shadows. The speeds estimated by the system for those cars totally accord with the velocities measured on video sequences by hand[†]. These results suggest that our method is also effective for estimating the speed of the vehicles in the shadows. Some practical traffic monitoring systems abandon the notion of discriminating the vehicles from shadows, however examples shown here give us an idea of how important discrimination of foreground objects from shadows is, so as to acquire accurate traffic information.

7. Conclusion

The objective of the work described in this paper is to apply our foreground-background-shadow segmentation method to the real problem of highway traffic surveillance, and to examine the validity of this method as a low-level car tracker through estimating velocities of cars. From this point of view, we have extended our method to car tracking and developed a basic highway surveillance system.

The contributions of this paper are summarized

[†] These sequences are provided without ground truth information.



field 730 field 732 field 734 field 736 field 738 field 740

field 1170 field 1172 field 1174 field 1176 field 1178 field 1180

Fig. 7 Two examples of tracking cars excluding shadows. Though the target cars on the right lane are within large shadows cast by the trucks on the left lane, they are reliably tracked over an area for test (denoted by a rectangle). Foreground: black, shadow: gray and background: white.

as follows. Firstly, we incorporated the context-dependence (formalized in Eq. (10)) among neighboring HMM regions at each time unit into state estimation (segmentation) phase. This not only promotes segmentation performance of the method, but makes it possible to find a foreground object as a connected region. This is also necessary for computing car speeds. Secondly, we designed a simple calibration that allows to calculate the speeds of vehicles while moving on the ground, using four point correspondences which are usually obtainable from the existence of road markers. Finally, several experiments utilizing car speedometers as ground truth confirmed the practical efficacy of our method, and the robustness of the method against the shadows.

Light conditions vary all the time. Our basic attitude toward this kind of alterations is to make HMMs suit the changing light conditions through updating the model parameters. This can be simply achieved by a re-learning process. However, the adaptiveness of the method for special time zones such as night, early morning, etc. or for inclement weather has not been tested. Problems caused by congested traffic conditions are also not discussed in this paper. These issues will be the subjects of future work.

The HMM-based segmentation method that plays important role in the system was initially developed as a low-level component for a high-level contour-based tracking process. Low-level approaches are fast and robust but convey little information other than object centroid. By "high-level" here, we mean that the trackers can follow complex deformations in high-dimensional spaces by including high-level shape and motion models [7]. Challenging further work will be to incorporate low-level information from our HMM-based component into a high-level contour-based tracker, under a consistent probabilistic framework. Introducing such a high-level foreground model will enhance the robustness of the system against the alterations from both light and weather conditions, and will contribute to solving various problems from congested traffic conditions.

References

[1] K.D. Baker and G.D. Sullivan, "Performance assessment of model-based tracking," Proc. IEEE Workshop on Applications of Computer Vision, pp.28–35, Palm Springs, CA, 1992.

[2] L.E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains," Ann. Math. Stat., vol.41, no.1, pp.164–171, 1970.

[3] A. Blake and M. Isard, Active contours, Springer, 1998.

[4] A.P. Dempster, N.M. Laird, and D.R. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," J.R. Stat. Soc., B 39, pp.1–38, 1977.

[5] O. Faugeras, Three-dimensional computer vision, The MIT Press, 2001.

[6] N. Ferrier, S. Rowe, and A. Blake, "Real-time traffic monitoring," Proc. Second IEEE Workshop on Applications of Computer Vision, pp.81–88, 1994.

[7] M. Isard and A. Blake, "ICONDENSATION: Unifying low-level and high-level tracking in a stochastic framework," Proc. 5th European Conf. on Computer Vision, pp.893–908, 1998.

[8] S. Kamijo, Y. Matsushita, K. Ikeuchi, and M. Sakauchi, "Occlusion robust tracking utilizing spatio-temporal Markov Random Field model," IEICE Trans. Inf. & Syst. (Japanese Edition), vol.J83-D-II, no.12, pp.2597–2609, Dec. 2000.

[9] J. Kato, T. Watanabe, and M. Yoneda, "HMM-based

back-ground-object-shadow separation for traffic monitoring movies," Trans. IPS Japan, vol.42, no.1, pp.1–15, 2001.

[10] J. Kato, T. Watanabe, S. Joga, J. Rittscher, and A. Blake, "An HMM-based segmentation method for traffic monitoring movies," IEEE Trans. Pattern Anal. & Mach. Intell., vol.24, no.9, pp.1291–1296, 2002.

[11] http://www.watanabe.nuie.nagoya-u.ac.jp/member/jien/demo.htm.

[12] J. Rittscher, First Year Report, The University of Oxford, Department of Engineering Science, 1999.

[13] D. Koller, K. Daniilidis, T. Thorhallson, and H. Nagel, "Model-based object tracking in traffic scenes," Proc. European Conference on Computer Vision, pp.437–452, Santa Margherita, Italy, May 1992.

[14] D. Koller, K. Daniilidis, T. Thorhallson, and H. Nagel, "Model-based object tracking in traffic scenes," Proc. European Conference on Computer Vision, pp.437–452, Santa Margherita, Italy, May 1992.

[15] D. Koller, J. Weber, and J. Malik, "Robust multiple car tracking with occlusion reasoning," Proc. European Conference on Computer Vision, pp.189–196, Stockholm, Sweden, 1994.

[16] J. Li, A. Najmi, and R.M. Gray, "Image classification by a two-dimensional hidden Markov model," IEEE Trans. Signal Processing, vol.48, no.2, pp.517–533, 2000.

[17] K. Maeda, K. Onoguchi, K. Fukui, and Y. Taniguchi, "Computer vision application to ITS," J. IEICE, vol.83, no.3, pp.191–195, 2000.

[18] J.L. Mundy and A. Zisserman, "Projective geometry for machine vision," in Geometric Invariance in Computer Vision, ed. J.L. Mundy and A. Zisserman, pp.463–519, MIT Press, Cambridge, MA, 1992.

[19] L.R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," Proc. IEEE, vol.77, no.2, pp.257–286, Feb. 1989.

[20] I.D. Reid, D.W. Murray, and K.J. Bradshaw, "Towards active exploration of static and dynamic scene geometry," IEEE Int'l Conference on Robotics and Automation, San Diego CA, 1994.

[21] S. Rowe and A. Blake, "Statistical mosaics for tracking," Image and Vision Computing, vol.14, pp.549–564, 1996.

[22] S. Takaba, "Significance of ITS and formation of its basic concept," J. IEICE, vol.83, no.7, pp.528–530, 2000.

Jien Kato received the M.E. and Ph.D. degrees in information engineering in 1990 and 1993, respectively, both from Nagoya University, Japan. From 1993 through 1999, she was with the Faculty of Engineering, Toyama University, Japan, where she worked primarily in the fields of document analysis and pattern recognition. In 1999, she joined the Robotics Research Group, the University of Oxford, and worked on visual tracking problems. She is currently an associate professor in the Department of Information Engineering, Graduate School of Engineering at Nagoya University. Dr. Kato is a member of the Information Processing Society of Japan and IEEE Computer Society.

Toyohide Watanabe received the B.S., M.S., and Ph.D. degrees from Kyoto University in 1972, 1974, and 1983, respectively. In 1987, he became an associate professor in Department of Information Engineering at Nagoya University. He is currently a professor in Department of Information Engineering, Graduate School of Engineering at Nagoya University. Dr. Watanabe's research interests include the knowledge/data engineering, computer supported collaborative learning, parallel and distributed process interaction, document understanding and drawing interpretation. He is a member of the Information Processing Society of Japan, Japan Society for Software Science, Japan Society of Artificial Intelligence, Japanese Society for Information and Systems in Education, ACM, AAAI, AACE, and IEEE Computer Society.

Hiroyuki Hase received his B.E. degree in electrical engineering from Toyama University in 1971, and Ph.D. degree from Tohoku University in 1989. He is associate professor of the Department of Intellectual Information Systems Engineering, Toyama University. His research interests are document analysis, character recognition and facial analysis. He is a member of the Information Processing Society of Japan, the Institute of Image Information and Television Engineers, the Institute of Image Electronics Engineers of Japan and IEEE computer society.