Traffic Monitoring and Accident Detection at Intersections

Shunsuke Kamijo, Yasuyuki Matsushita, Katsushi Ikeuchi, *Fellow, IEEE*, and Masao Sakauchi

***Abstract*—Among the most important research in Intelligent Transportation Systems (ITS) is the development of systems that automatically monitor traffic flow at intersections. Rather than being based on global flow analysis as is currently done, these automatic monitoring systems should be based on local analysis of the behavior of each vehicle at the intersection. The systems should be able to identify each vehicle and track its behavior, and to recognize situations or events that are likely to result from a chain of such behavior. The most difficult problem associated with vehicle tracking is the occlusion effect among vehicles. In order to solve this problem we have developed an algorithm, referred to as spatio-temporal Markov random field (MRF), for traffic images at intersections. This algorithm models a tracking problem by determining the state of each pixel in an image and its transit, and how such states transit along both the x–y image axes as well as the time axes. Vehicles, of course, are of various shapes and they move in random fashion, thereby leading to full or partial occlusion at intersections. Despite these complications, our algorithm is sufficiently robust to segment and track occluded vehicles at a high success rate of 93%–96%. This success has led to the development of an extendable robust event recognition system based on the hidden Markov model (HMM). The system learns various event behavior patterns of each vehicle in the HMM chains and then, using the output from the tracking system, identifies current event chains. The current system can recognize bumping, passing, and jamming. However, by including other event patterns in the training set, the system can be extended to recognize those other events, e.g., illegal U-turns or reckless driving. We have implemented this system, evaluated it using the tracking results, and demonstrated its effectiveness.**

***Index Terms*—Accident detection, hidden Markov model (HMM), occlusion, spatio-temporal Markov random field (MRF), tracking.**

I. Introduction

ONE of the most important research efforts in Intelligent Transportation Systems (ITS) is the development of systems that automatically monitor the flow of traffic at intersections. Such systems would be useful both in reducing the workload of human operators and in warning drivers of dangerous situations. Not only would the systems automatically monitor current situations at intersections but, if the systems could reliably assess those situations and predict whether they might lead to accidents, they might be able to warn drivers and thus reduce the number of accidents.

Rather than the current practice of performing a global flow analysis, the automatic monitoring systems should be based on local analysis of the behavior of each vehicle at intersections. The systems should be able to identify each vehicle and track its behavior, and to recognize dangerous situations or events that might result from a chain of such behavior. Tracking at intersections is often impeded by the occlusion that occurs among vehicles in crowded situations. Also, event recognition may be complicated by the large variations in a chain of events. To solve these problems, it is necessary to develop a tracking algorithm that is robust against such occlusion. In addition, stochastic robust event recognition must also be developed.

Tracking algorithms have a long history in computer vision research. In particular, in ITS areas, vehicle tracking, one of the specialized tracking paradigms, has been extensively investigated. Peterfreund [1] employs the “Snakes” [2] method to extract contours of vehicles for tracking purposes. Smith [3] and Grimson [4] employ optical-flow analysis. In particular, Grimson applies clustering and vector quantization to estimated flows. Leuck [5] and Gardner [6] assume three-dimensional (3-D) models of vehicle shapes, and estimated vehicle images are projected onto a two-dimensional (2-D) image plane according to appearance angle. The methods of Leuck [5] and Gardner [6] require that many 3-D models of vehicles be applied to general traffic images. While these methods are effective in less crowded situations, most of them cannot track vehicles reliably in situations that are complicated by occlusion and clutter.

Our major goal is to track individual vehicles robustly against the occlusion and clutter effects which usually occur at intersections. Vehicles traveling through intersections are moving in various directions; various parts of these vehicles may either be occluded by, or themselves occlude, other vehicles. In order to overcome such situations, we have developed a tracking algorithm utilizing the spatio-temporal Markov random field (MRF) model. This algorithm models a tracking problem by determining the state of each pixel in an image, and how the states transit along both the x–y image axes and the time axes.

For event recognition, systems traditionally use spot sensors. Successful event recognition systems with spot sensors for traffic monitoring include Gangisetty’s [8] Incident Detection System (IDS) with inductive loop sensors. Traffic monitoring system using loop detectors are quite popular and, in fact, in practical use in several cities including Toronto, ON, Canada and Washington, DC. Although such spot sensors are reliable, stable, and in practical use, they have rather limited scope of usage in terms of event recognition; almost all of the spot sensors can obtain information only on whether a vehicle

Manuscript received March 7, 2000; revised September 15, 2000. The Guest Editor for this paper was Prof. Charles E. Thorpe.

The authors are with the Institute of Industrial Science, University of Tokyo, 7-22-1 Roppongi, Minato-ku, Tokyo 106-8558, Japan (e-mail: kamijo@sak.iis.u-tokyo.ac.jp).

Publisher Item Identifier S 1524-9050(00)10461-2.

exists on a sensor spot; therefore, a large number of sensors are required to survey an area for event recognition.

On the other hand, one of the most important advantages of utilizing vision sensors for event recognition is their ability to collect rich information such as illegally parked vehicles, traffic jams, traffic violations, and accidents. Some representative vision-based systems can be found in [11]–[13], [4]. Rojas [11] and Zeng [12] developed methods or employed systems for tracking vehicles on highways from a fixed TV camera. Lai *et al.* [13] developed "Red light runners detection" at an intersection. This system is now in operation in Hong Kong. Grimson, *et al.* [4] are monitoring traffic by "Forest of Sensors." The traffic activity is classified by clustering motion vectors and detecting abnormal events.

Unfortunately, however, these systems have rather limited capability to detect events. For example, the Hong Kong system can recognize only red-light runners and cannot be extended to other activities. We have derived an extendable robust event recognition system based on the hidden Markov model (HMM). The system learns various event patterns of behavior of each vehicle in the HMM chains, and then identifies current event chains using the output from the tracking system. The current system can recognize bumping, passing, and jamming. However, by including other event patterns in the training set, the system can be extended to recognize them, e.g., illegal U-turns or reckless driving.

In this paper, we will first describe the occlusion tracking algorithm that utilizes the spatio-temporal Markov random field (ST-MRF) model in Section II. We will also show an experimental result of this algorithm using sequences of intersection images. Then, we will describe our system's accident detection method in Section IV. This method is based on the HMM. Input to the system is from time series observation of behaviors of individual vehicles acquired by the tracking method described in Section II.

II. Occlusion Robust Tracking Algorithm Utilizing Spatio-Temporal MRField Model

A. Basic Ideas

We will employ a stochastic relaxation algorithm for vehicle tracking. Because vehicles vary in their appearance, shape-based tracking is less effective for tracking; and, because vehicles travel in cluttered intersections where occlusions often occur, they cannot be tracked by a simple contour-based method. To overcome these problems, we developed a dedicated tracking algorithm based on a stochastic relaxation algorithm. We can model a tracking problem as a labeling problem to each pixel in an image sequence, whether a pixel is assigned to vehicle A or vehicle B. These labels can be considered to transit or to have some relation to each other along both the time and the spatial x–y image axes. Thus we can model this transition as an MRFmodel along the time and spatial axes. We refer to this model as ST-MRF model. Our tracking algorithm is designed using this ST-MRF model to determine labels at each position in images.

In preparation for this algorithm, an image which consists of 640 × 480 pixels is divided into 80 × 60 blocks because a pixel is too small to be considered as one site in the ST-MRF and, therefore, we need a larger group of pixels. Each block consists of 8 × 8 pixels. One block is considered to be a site in the ST-MRF. The algorithm classifies each block into vehicles or, equivalently, assigns one vehicle label to each block. Since the ST-MRF converges rapidly to a stable condition when it has a good initial estimation, we use a deductive algorithm to determine an initial label distribution. Then, these labels are refined through the ST-MRF. In this refining process, the algorithm considers correlation of blocks between consecutive images as well as neighbor blocks and then assign labels to them through the MRF model. A distribution of classified labels on blocks is referred to as an object map.

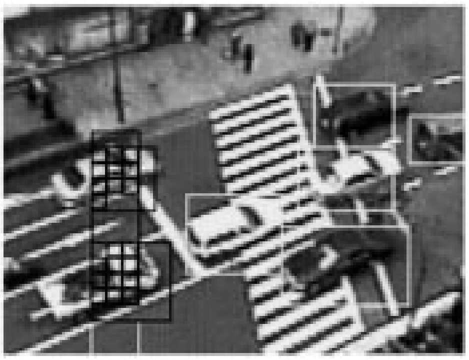


Fig. 1. Object generation.

Section II-B briefly explains the deductive method for obtaining an initial object map. Then, Sections II-C–II-E describe a refinement method based on the ST-MRF.

B. Deductive Process of Tracking Algorithm

Here we describe a tracking method adapted to variances in size and shape of vehicles. The output from this method will be used as an initial estimation of an object map for the ST-MRF tracking method.

[Deductive Process for Tracking Algorithm]

1) Initialization (Fig. 1): A background image of the intersection is constructed by accumulating and averaging an image sequence for a certain interval of time, in our case, 20 min. The algorithm also sets up a slit at each entrance to the intersection for examining incoming vehicles. Here, a slit is one line of an image segment to determine intensity variance, for example, four slits perpendicular to the four major incoming roads to the intersection are set up in Fig. 1.
2) Generate new vehicle IDs (Fig. 1): An incoming vehicle is detected when intensity difference occurs at a block along a slit. At each slit block, differences between the current and background intensities are examined, and, if the differences are greater than a certain threshold value, the algorithm decides that it detects a vehicle, generates a new vehicle ID, and assigns it to the block. Along the movement of the vehicle, the slit continuously detects the vehicle and continues to issue the same vehicle ID. Blocks having the same vehicle ID grow along the time sequence.

Those blocks sharing the same object ID are grouped into a vehicle region.

3) Estimate Motion Vectors of Vehicles: Once a vehicle region leaves the slit, the algorithm updates its shape along the time sequence. For this updating, the algorithm estimates motion vectors among blocks in a vehicle region. At each block, a block matching method is employed to estimate its motion vector between the time t and $t+1$. Similarity between one block at time t at $(x(t),\, y(t))$ and one at time $t+1$ at

$$(x(t+1),\, y(t+1)) = (x(t)+u(t),\, y(t)+v(t))$$

is evaluated as (1); here, $I(x,\, y;\, t)$ is a gray-scaled intensity of a pixel $(x,\, y)$ at time t. Then the motion vector of a vehicle is approximated by the most frequent motion vector among those blocks of the same vehicle region.

$$D = \sum_{0\le di<8,\, 0\le dj<8} \left| I\left(i+di+v_i,\, j+dj+v_j;\, t+1\right) - I\left(i+di,\, j+dj;\, t\right) \right|. \quad (1)$$

4) Update Vehicle Regions (Fig. 2): By using the motion vector, all the blocks of the same vehicle region shift at the time $t+1$ from the locations at the time t. After a block has shifted, if the intensity difference between the current and the background at the new location is smaller than a threshold value, the algorithm does not consider the block as belonging to the vehicle region. On the other hand, if a neighbor block of the new vehicle region has a larger difference, the algorithm defines the block to be the vehicle region, and assigns the same vehicle ID.

5) Divide Vehicle Blocks: In some cases, multiple vehicles simultaneously pass through a slit and they may be considered to be a single object. In order to divide such vehicles thereafter, the algorithm examines connectivity and the distribution of motion vectors over object blocks. If multiple parts in one vehicle region have several different motion vectors, those parts are separated and assigned to different vehicle IDs.

The result obtained in this subsection will be utilized in Section II-C as the initial object map for the stochastic relaxation process based on the ST-MRF model. Because a motion vector is estimated by a measure of pixels, not blocks, in Step 3), a fragmentation problem will occur in renewing the Object-Map. When a block is considered to belong to two different blocks in a step, it must be determined to which object it likely belongs. These two problems can be optimized by the stochastic relaxation method, S-T MRF.

C. Spatio-Temporal MRF Model

Some blocks may be classified as having multiple vehicle labels due to occlusion and fragmentation. We can resolve this ambiguity by employing stochastic relaxation with the MRF model. Several representative research efforts exist in computer vision, including image restoration by Geman and Geman [15]; image compression by Chellapa, Chatterjee, and Bargdzian [16]; and image segmentation by Andrey and Tarroux [17]. Usually, an MRF model handles only spatial x-y directional distribution, i.e., an image. We extend it to be able to handle not only spatial distribution but also time-axis distribution. An image sequence has correlations at each pixel between consecutive images along a time axis. Our MRF also considers this time-axis correlation. We named the extended MRF the (spatio-temporal) ST-MRF model.

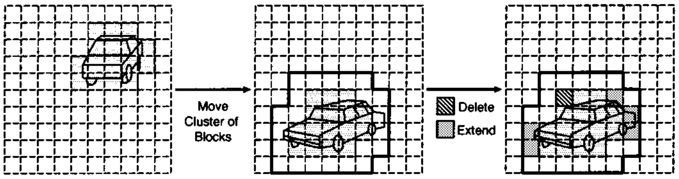


Fig. 2. Object moving and remapping.

Our ST-MRF estimates a current object-map (a distribution of vehicle labels) according to a previous object map, and previous and current images. Here are the notations:

- $G(t-1)=g,\, G(t)=h$: An image G at time $t-1$ has a value g, and G at time t has a value h. At each pixel, this condition is described as
$$G(t-1;\, i,\, j) = g(i,\, j) \qquad G(t;\, i,\, j) = h(i,\, j).$$
- $X(t-1)=x,\, X(t)=y$: An object Map X at time $t-1$ is estimated to have a label distribution as x, and X at time t is estimated to have a label distribution as y. At each block, this condition is described as
$$X_k(t-1) = x_k \qquad X_k(t) = y_k$$
where k is a block number.

We will determine the most likely $X(t)=y$ so as to have the MAP (Maximum *A posteriori* Probability) for given $G(t-1)=g,\, G(t)=h$, and $X(t-1)=x$ previous and current images and a previous object map, and a previous object map $X(t)=y$.*A posteriori* probability can be described using the Bayesian equation

$$\begin{aligned} &P(X(t)=y|G(t-1)=g,\, X(t-1)=x,\, G(t)=h) \\ &= \frac{\begin{array}{c} P(G(t-1)=g,\, X(t-1)=x,\, G(t)=h|X(t)=y) \\ \cdot P(X(t)=y) \end{array}}{P(G(t-1)=g,\, X(t-1)=x,\, G(t)=h)}. \end{aligned} \quad (2)$$

$P(G(t-1)=g,\, X(t-1)=x,\, G(t)=h)$, a probability to have previous and current images and a previous object map, can be considered as a constant. Consequently, maximizing *a posteriori* probability is equal to maximizing $P(G(t-1)=g,\, X(t-1)=x,\, G(t)=h|X(t)=y)P(X(t)=y)$.

$P(X(t) = y)$ is a probability for a block C_k to have $X_k(t-1) = y_k$ (for all ks). Here, y_k is a vehicle label. For each C_k, we can consider its probability as a Boltzmann distribution. Then, $P(X(t)=y)$ is a product of these Boltzmann distributions

$$\begin{aligned} P(X(t)=y) &= \prod_k \exp\left[-U_N\left(N_{y_k}\right)\right]/Z_{Nk} \\ &= \prod_k \exp\left[-\frac{1}{2\sigma_{N_y}^2}\left(N_{y_k}-\mu_{N_y}\right)^2\right] \Bigg/ Z_{Nk} \end{aligned} \quad (3)$$

Here N_{y_k} is the number of neighbor blocks of a block C_k (Fig. 3) that belong to the same vehicle as C_k. Namely, the

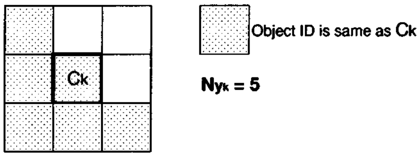


Fig. 3. Eight neighbor blocks.

more neighbor blocks that have the same vehicle label, the more likely the block is to have the vehicle label. Currently, we consider eight neighbors as shown in Fig. 3. Thus $\mu_{N_y} = 8$, because the probability related to block C_k has maximum value when block C_k and all its neighbors have the same vehicle label. Therefore, the energy function $U_N(N_{y_k})$ takes a minimum value at $N_y = 8$ and a maximum value at $N_y = 0$.

We also consider the probability of $G(t-1)=g$, $G(t)=h$, $X(t-1)=x$ for a given object map $X(t) = y$ as a Boltzmann function of two independent variables

$$\begin{aligned} &P(G(t-1) = g,\, X(t-1) = x,\, G(t) = h | X(t) = y) \\ &\quad = \prod_k \exp\left[-U_{pre}\left(M_{xy_k},\, D_{xy_k}\right)\right] / \, Z_{DMk} \\ &\quad\quad \cdot \prod_k \exp\left[-U_M\left(M_{xy_k}\right)\right] / \, Z_{Mk} \\ &\quad = \prod_k \exp\left[-U_D\left(D_{xy_k}\right)\right] / Z_{Dk} \\ &\quad = \prod_k \exp\left[-\frac{1}{2\sigma^2_{M_{xy}}}\left(M_{xy_k} - \mu_{M_{xy}}\right)^2\right] \Bigg/ \, Z_{Mk} \\ &\quad\quad \cdot \prod_k \exp\left[-\frac{1}{2\sigma^2_{D_{xy}}}\left(D_{xy_k} - \mu_{D_{xy}}\right)^2\right] \Bigg/ \, Z_{Dk}. \end{aligned} \tag{4}$$

M_{xy_k} is a goodness measure of the previous object map $X(t-1) = x$ under a given current object map $X(t) = y$. Let us assume that a block C_k has a vehicle label O_m in the current object map $X(t)$, and C_k is shifted backward in the amount of estimated motion vector, $-\overrightarrow{V_{O_m}} = (-v_{mi},\, -v_{mj})$ of the vehicle O_m, in the previous image (Fig. 4). Then the degree of overlapping is estimated as M_{xy_k}, the number of overlapping pixels of the blocks with the same vehicle labels. The more pixels that have the same vehicle label, the more likely a block C_k belongs to the vehicle. The maximum number is $\mu_{M_{xy}} = 64$, and the energy function $U_M(M_{xy_k})$ takes a minimum value at $M_{xy_k} = 64$ and a maximum value at $M_{xy_k} = 0$.

For example, when a block is determined to which of vehicle O_1, O_2 it belongs, $U_M(M_{xy_k})$ will be estimated as follows. First, assuming that a block belongs to O_1, the energy function is estimated as $U_M(M_{xy_k}) = U_{M1}$ by referring to $-\overrightarrow{V_{O_1}} = (-v_{1i},\, -v_{1j})$. Then assuming that a block belongs to O_2, the energy function is estimated as $U_M(M_{xy_k}) = U_{M2}$ by referring to $-\overrightarrow{V_{O_2}} = (-v_{2i},\, -v_{2j})$. As result of these estimations, when U_{M1} is less than U_{M2}, this block more likely belongs to vehicle O_{M1}.

D_{xy_k} represents texture correlation between $G(t-1)$ and $G(t)$. Let us suppose that C_k is shifted backward in the image $G(t-1)$ according to the estimate motion vector $-\overrightarrow{V_{O_m}} = (-v_{mi},\, -v_{mj})$. The texture correlation at the block C_k is evaluated as (see Fig. 5)

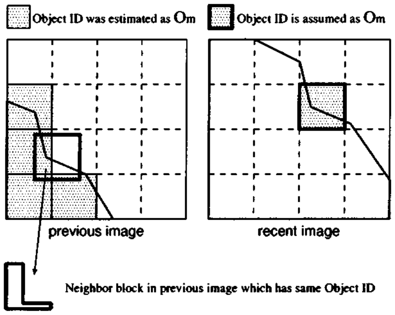

Object ID was estimated as Om
Object ID is assumed as Om
previous image
recent image
Neighbor block in previous image which has same Object ID

Fig. 4. Neighbor condition between consecutive images.

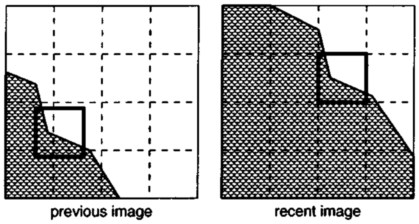


Fig. 5. Texture matching.

$$\begin{aligned} D_{xy_k} = \sum_{0 \le di < 8,\, 0 \le dj < 8} &\Big| G(t;\, i + di,\, j + dj) \\ &- G(t-1;\, i + di - v_{mi},\, j + dj - v_{mj}) \Big|. \end{aligned} \tag{5}$$

The energy function $U_D(D_{xy_k})$ takes maximum value at $D_{xy_k} = 0$. The smaller D_{xy_k} is, the more likely C_k belong to the vehicle. That is, the smaller $U_D(D_{xy_k})$ is, the more likely C_k belong to the vehicle.

For example, when a block is determined to which of vehicle O_1, O_2 it belongs $U_D(D_{xy_k})$ will be estimated as follows. First, assuming that a block belongs to O_1, the energy function is estimated as $U_D(D_{xy_k})) = U_{D1}$ by referring to $\overrightarrow{V_{O_1}} = (v_{1i},\, v_{1j})$. Then assuming that a block belongs to O_2, the energy function is estimated as $U_D(D_{xy_k})) = U_{D2}$ by referring to $\overrightarrow{V_{O_2}} = (v_{2i},\, v_{2j})$. As a result of these estimations, when U_{D1} is less than U_{D2} this block most likely belongs to vehicle O_{D1}.

Consequently, this optimization problem results in a problem of determining a map $X(t) = y$ which minimizes the following energy function:

$$U(y) = \sum_k U\left(y_k\right) \tag{6}$$

$$\begin{aligned} U(y_k) &\equiv U_N\left(N_{y_k}\right) + U_{pre}\left(D_{xy_k},\, M_{xy_k}\right) \\ &= U_N\left(N_{y_k}\right) + U_D\left(D_{xy_k}\right) + U_M\left(M_{xy_k}\right) \\ &= a\left(N_{y_k} - \mu_{N_y}\right)^2 + b\left(M_{xy_k} - \mu_{M_{xy}}\right)^2 + cD^2_{xy_k}. \end{aligned} \tag{7}$$

$U(y_k)$ is considered to be the energy function for ST-MRF, and $U(y_k)$ will be minimized by the relaxation process.

D. Relaxation Algorithms

In the stochastic relaxation algorithms, Gibbs Sampler and Metropolis algorithms are often used. The Gibbs Sampler algorithm considers all states as the candidate states in the next step of the relaxation loop process. Then it evaluates the probabilities of the states according to the Boltzmann distribution function. The algorithm then determines which of the states to transit to. On the other hand, the Metropolis algorithm randomly selects one state among all possible candidate states, and then determines whether a transition to the state occurs (the block keeps the current state) according to the Boltzmann distribution function. Both algorithms represent a probability for a block C_k to have a state y_k as $(U(y_k)/Z_k)$. Here, a state means a vehicle label in our application.

E. Implementation of Relaxation Processes

Consider a situation where a block was considered to belong to two different vehicles, as described in Section II-A; here, we want to determine to which vehicle the block is likely to belong, that is, we want to determine which label the block is likely to have.

We apply the ST-MRF and relaxation process only to those confused blocks. Under this formulation, the energy function can be approximated to be a simpler differential form

$$\begin{aligned} U_{12}(y_k) \equiv\, & U(y_k=O_1) - U(y_k=O_2) \\ =\, & a\left[\left(N_{y_k=O_1}-\mu_{N_y}\right)^2 - \left(N_{y_k=O_2}-\mu_{N_y}\right)^2\right] \\ & + b\left[\left(M_{xy_k=O_1}-\mu_{M_{xy}}\right)^2 - \left(M_{xy_k=O_2}-\mu_{M_{xy}}\right)^2\right] \\ & + c\left[D^2_{xy_k=O_1} - D^2_{xy_k=O_2}\right]. \end{aligned} \tag{8}$$

Here, $U(y_k = O_1)$ is the energy for a block C_k to have a vehicle label, vehicle O_1 among the two possible labels, O_1 and O_2. By a simple derivation, energy equation (8) can be written as

$$\begin{aligned} U_{12}(y_k) =\, & a\left[N_y(y_k=O_1) - N_y(y_k=O_2)\right] \\ & \cdot \left[N_y(y_k=O_1) + N_y(y_k=O_2) - 2\mu_{N_y}\right] \\ & + b\left[M_{xy}(y_k=O_1) - M_{xy}(y_k=O_2)\right] \\ & \cdot \left[M_{xy}(y_k=O_1) + M_{xy}(y_k=O_2) - 2\mu_{M_{xy}}\right] \\ & + c\left[D_{xy}(y_k=O_1)^2 - D_{xy}(y_k=O_2)^2\right]. \end{aligned} \tag{9}$$

Since we are trying to apply this energy function to confused blocks, block C_k is usually surrounded by blocks of two possible labels. Therefore, it is regarded as follows:

$$N_y(y_k=O_1) + N_y(y_k=O_2) = \mu_{N_y} \tag{10}$$

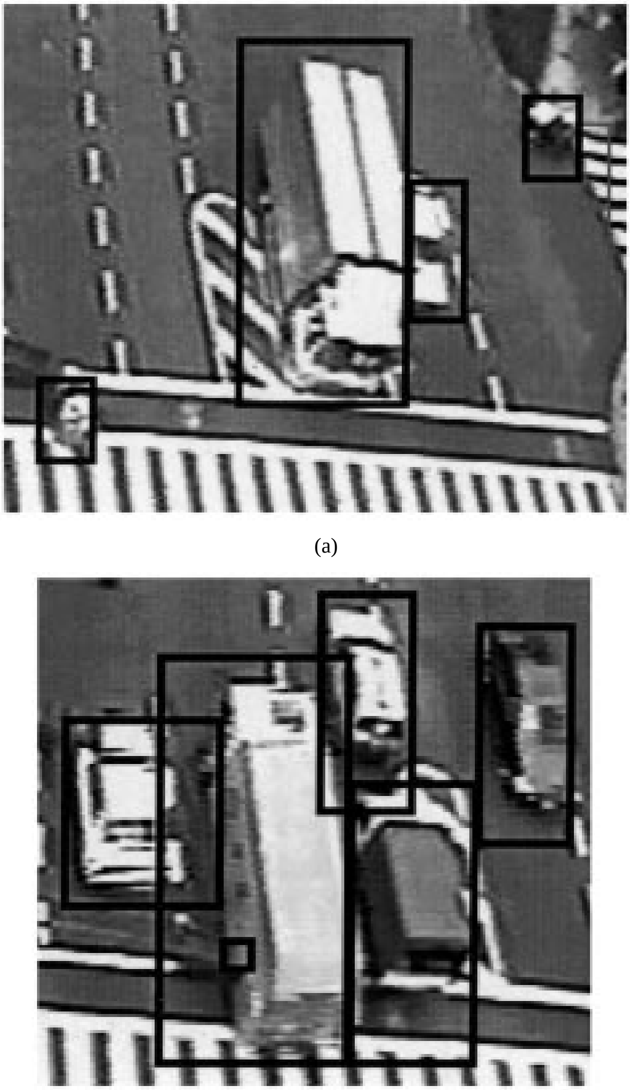
$$M_{xy}(y_k=O_1) + M_{xy}(y_k=O_2) = \mu_{M_{xy}}. \tag{11}$$

Consequently, the procedure results in the following energy function:

$$\begin{aligned} U_{12}(y_k) =\, & -\alpha\left[N_y(y_k=O_1) - N_y(y_k=O_2)\right] \\ & - \beta\left[M_{xy}(y_k=O_1) - M_{xy}(y_k=O_2)\right] \\ & + \gamma\left[D_{xy}(y_k=O_1)^2 - D_{xy}(y_k=O_2)^2\right] \\ & \cdot (\alpha 0,\, \beta 0,\, \gamma 0). \end{aligned} \tag{12}$$

Equation (12) is symmetrical with respect to O_1 and O_2. If $U_{12}(y_k)$ is negative, the block C_k is more likely to have O_1 than O_2. On the other hand, if it is positive, the block C_k is more likely to have O_2 than O_1.

The relaxation process for block C_k can be described as a loop of the following procedure: at first, estimate differential energy $U_{12}(y_k)$, in the second, select O_1 or O_2 as the next state for C_k according to $U_{12}(y_k)$ using the Metropolis algorithm.


(a)

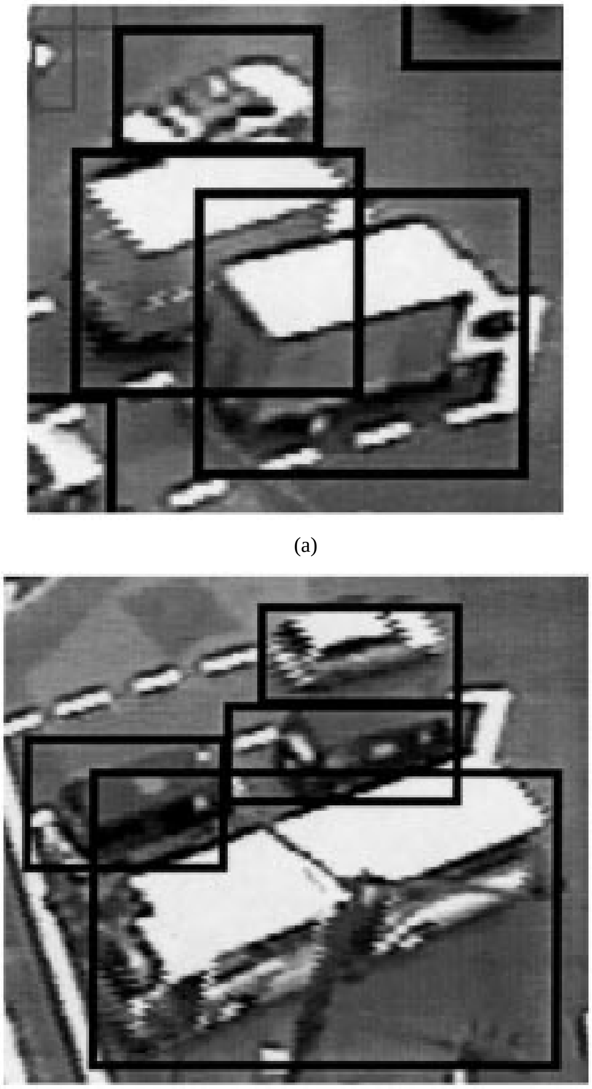
(b)

Fig. 6. Tracking results against occlusion and confusion. Vertical traffic. (a) Occlusion. (b) Clutter.

F. Experimental Results

We applied the tracking algorithm utilizing the ST-MRF model to 25-min traffic images at the intersection. We used 256-level gray-scaled image, and conditions were $\alpha 1.0$, $\beta = 0.125$, $\gamma = 1/4\,000\,000$. Three thousand two hundred and fourteen vehicles traversed the intersection; of these, 541 were occluded. As a result, the method was able to track separated vehicles that did not cause occlusions at over 99% success rate, and the method was able to segment and track 541 occluded vehicles at about 95% success rate. Detailed success rates and results of tracking in the occlusion situations are shown in Table I and Figs. 6 and 7.

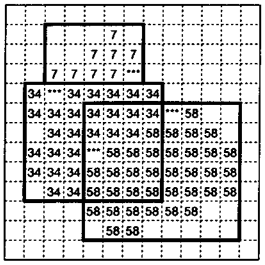
Figs. 6 and 7 show results of tracking in occlusion and clutter situations, and Fig. 8 shows an Object-Map for Fig. 7(a). The numbers, 7, 34, 58 mean attached ID to the three vehicles, and *** signifies a confusion block; it cannot be determined



(a)

(b)

Fig. 7. Tracking results against occlusion and confusion: Horizontal traffic. (a) Occlusion. (b) Clutter.

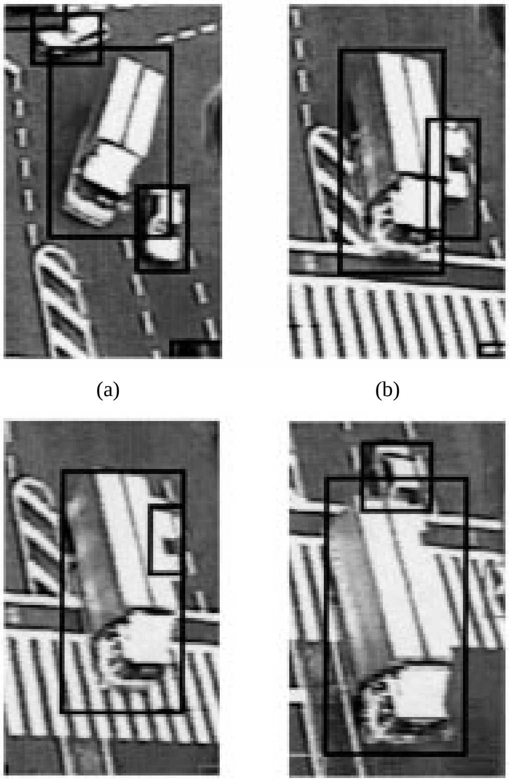


					7							
				7	7	7						
		7	7	7	7	***						
	34	***	34	34	34	34	34					
	34	34	34	34	34	34	34	***	58			
		34	34	34	34	34	58	58	58	58		
	34	34	34	***	58	58	58	58	58	58	58	
	34	34	34	58	58	58	58	58	58	58	58	
		34	34	58	58	58	58	58	58	58	58	
				58	58	58	58	58	58			
					58	58						

Fig. 8. Object-Map of occluded vehicles.

to which vehicle it belongs. Although rectangle areas seem to be overlapping in Fig. 7(a), it is merely for the sake of the drawing; the algorithm successfully separated those vehicles.

Fig. 9 shows a sequence of tracking two vehicles in the case of Fig. 6. These images are obtained at the rate of 10 frames/s, and a frame number is attached to each image. Although a car is partly occluded behind a truck, the two vehicles have been successfully segmented. Fig. 10 shows the Object-Maps corresponding to images of Fig. 9.



(a) (b)

(c) (d)

Fig. 9. Tracking results by ST-MRF. (a) Frame 678. (b) Frame 696. (c) Frame 702. (d) Frame 710.

Table I shows success rates of tracking vehicles without occlusions. As shown in this table, the usefulness of our tracking algorithm has been proven to be effective. Table I includes complicated situations that cannot be categorized either for horizontal or vertical traffic. Here, "horizontal traffic" means the traffic which travels along horizontal direction of image, and "vertical traffic" means the traffic which travels vertical direction of image. The major reason for the errors is due to the complete occlusion. In such cases, occluded vehicles have no block of their textures. Such situations often occur in horizontal traffic.

III. Traffic Monitoring System

A. System Review

In order to observe traffic activity at a particular intersection, we have set up a video camera on the roof of a building overlooking that intersection. Analog colored video images, obtained by the camera, are transferred to our laboratory through the NTT optical fiber line. The images are divided into two channels in our laboratory. One channel is used for recording purposes and directly leads to an S-VHS recorder; another channel is used for generating background images in real time. The images in this channel are captured through SGI video capture board, and digitized in real time for the purpose of composing a background image. Currently, accident detection is done off-line

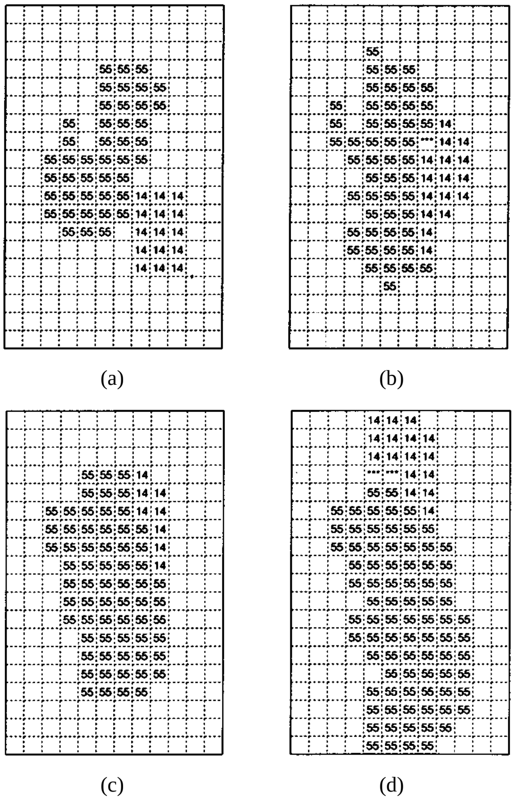


Fig. 10. Object-Map by ST-MRF. (a) Frame 678. (b) Frame 696. (c) Frame 702. (d) Frame 710.

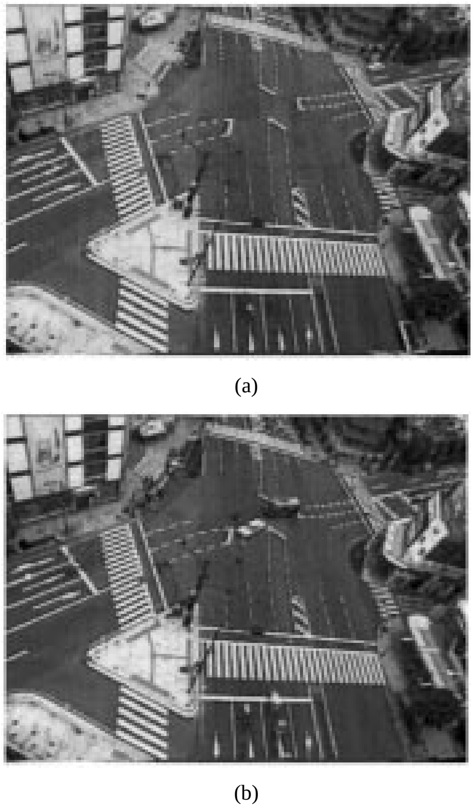


Fig. 11. Background changing due to an accident. (a) Background image before an accident. (b) Background image after an accident.

TABLE I
TRACKING RESULTS IN OCCLUSION SITUATIONS

	Horizontal Traffic	Vertical Traffic	Total
Number of Tracked Vehicles	305	236	541
Number of Tracked Vehicles successfully	285	227	521
Success Rate	93.0%	96.8%	94.6%

due to the limited performance of SGI-O2, while background images are being generated in real time.

B. Acquiring Background Images

Background images are composed from a sequence of images for a 20-min duration at intervals of every 10 min. Thus consecutive two background images overlap each other by 10 min.

A background image is generated through histogram analysis of each pixel along an image sequence. The algorithm makes a histogram of intensities at each pixel along an image sequence. Here, each image sequence is currently given for a period of 20 min. Then, the algorithm determines the intensity of the maximum frequency in the histograms. The maximum-frequency intensity is assigned to each pixel for a background image.

For clear background images, a longer duration of accumulating intensities is desirable in histogram making. On the other hand, for frequent detection of abnormal events from background images, a shorter duration is desirable. Considering this tradeoff, we have decided on a 20-min duration for the accumulation so as to be able to ignore vehicles that stop at red lights. The 10-min interval for renewing background images is determined so as to enable detection of illegally parked vehicles (currently defined in traffic law as those staying for more than 10 min). Considering a case of an accident, it usually takes more than 30 min from the instant an accident occurs to the end of an inspection by the police. During this period, every 10 min a new event occurs such as the arrival of police cars or ambulance. We have decided to update the background every 10 min to detect these new events. See an example of finding an accident from a background image in Fig. 11.

The composed background can ignore slower variations due to weather changes in an image sequence. It gradually accumulates images so as to average small continuous variance. On the other hand, it is effective for finding rapid variations caused by illegally parked vehicles or obstacles dropped from run-away vehicles. These rapid changes are detected as differences between two consecutive background images.

Background change is also useful for detecting accidents. Generally speaking, vehicles responsible for an accident remain nearby the intersection for a while. And shortly after the accident, ambulances and police cars also arrive; these also stay at the accident scene for a while. Such phenomena account for the differences among background images through several 10-min periods as shown in Fig. 11. Therefore, these background images are useful for retrieving images previously recorded. As mentioned above, our system records image sequences on an S-VHS recorder in addition to composing background images on-line. By scanning background images, we are able to find an

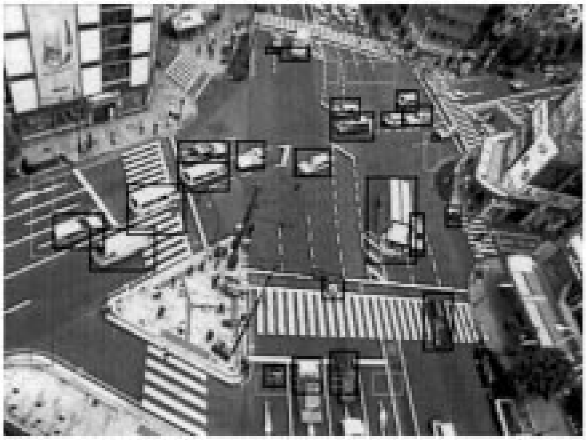


Fig. 12. The entire image of the intersection.

image corresponding to an abnormal event. Then we can retrieve an image sequence recording that abnormal event on S-VHS tapes.

By using a database obtained in this way, we are developing algorithms for traffic monitoring and abnormal event detection, such as accident detection, which is described in Section IV.

IV. Accident Detection Utilizing HMM

Fig. 12 shows an entire image of the intersection which was observed through all the days. Using tracking results obtained in Section II, we are trying to analyze traffic images and detect abnormal events such as accidents. Although the detection algorithm is still under evaluation, we would like to introduce a method for accident detection.

A. Feature Extraction

By using the tracking results, we developed a method for accident detection which can be generally adapted to intersections of any geometry. We first describe the dedicated feature extraction algorithm in this section, and the algorithm to detect accidents using HMM in the next section.

We designed our accident detection algorithm so as to be independent of geometric factors, such as geometry of the intersection, angle of video camera, and position where the accident occurred. Yamato [19] shows successful results in gesture recognition of a tennis player. In that paper [19], images themselves were divided into blocks and observations were extracted directly from images within blocks. Though successful, positions of a video camera and tennis player are supposed to be fixed. Considering our purpose, vehicles run in so many ways and accidents may occur everywhere in an image frame. Therefore, such geometric dependencies increase the amount of training data for such supervised learning methods as HMM. Thus we intentionally avoid using image intensity itself for the features, because it depends on the color of vehicles. We also avoid employing motion vectors themselves with the features.

We defined features for HMM as follows (see Fig. 13). The following algorithm and recognition by HMM are applied to every pair of objects.

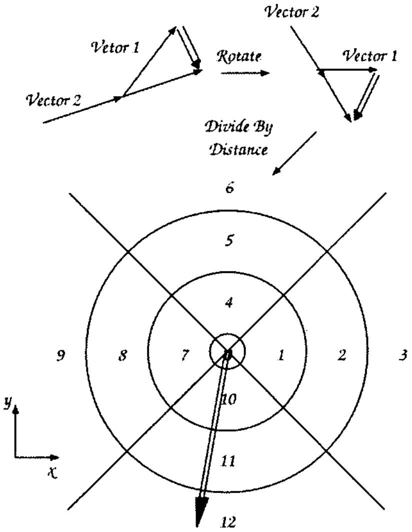


Fig. 13. Dedicated feature extraction.

1) Estimate the difference of motion vectors between the two objects

$$\vec{V}_d = \vec{V}_1 - \vec{V}_2. \tag{13}$$

2) Rotate the differential motion vector so as to align Vector-1 to X-plus direction

$$\vec{V}_n = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \vec{V}_d. \tag{14}$$

3) Divide the rotated vector by the distance between the two objects. Here, the distance is measured by the nearest blocks between the two objects, because the nearest distance directly affects the collision time of the two objects

$$\vec{V}_n = \vec{V}_r / d_{12}. \tag{15}$$

4) Quantize the rotated vector as an input observation 0–12 for HMM. Even if the differential vector is the same, the degree of danger depends to a great extent on the distance. So, a larger norm of $\overrightarrow{V_n}$ means a more dangerous situation, that is observations 3, 6, 9, and 12.

B. Recognition Method

Accident detection is a class of recognition problems to classify time series observations. There are various techniques for time-series event recognition, such as DP-Matching, Neural Network (NN), and hidden Markov model (HMM).

At first, we prefer methods based on stochastic models to those based on concrete models, because an accident sequence consists of a large number of random processes and those can be described by using stochastic models. DP-Matching is a kind of strict recognition method which is not robust against disturbance in observations. On that point, stochastic methods utilizing HMM or NN are superior to DP-matching.

In addition, we prefer methods that are robust against disturbance in length of observation sequences. HMM is considered to be superior to NN on that point.

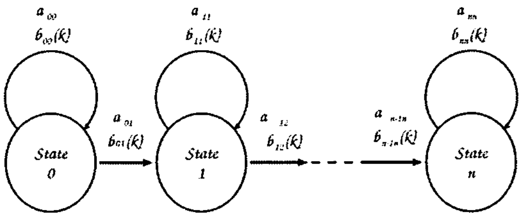


Fig. 14. Left-to-right HMM.

C. Accident Detection Utilizing HMM

Accident detection is a class of recognition problems to handle time-series events. There are various techniques for time-series event recognition, such as DP-Matching, Neural Network, and Hidden Markov Model. As stated earlier, we prefer methods based on stochastic models to those based on concrete models, because an accident sequence consists of a large number of random processes and those can be described by using stochastic models. Thus we decided to apply HMM to accident detection.

We applied simple left-to-right HMM for accident detection, as shown in Fig. 14. An HMM model has the following parameters:

- a_{ij}: Transition probability from state i to state j, where $a_{ij} = 0 (j \neq i, i + 1)$;
- $b_{ij}(k)$: Probability to output observation k when transition from state i to state j occurred, where $b_{ij}(k) = 0 (j \neq i, i + 1)$.

Parameters a_{ij}, $b_{ij}(k)$ are trained by the Baum–Welch algorithm [20], [21]. Here we define forward variables $\alpha_t(i)$ and backward variables $\beta_t(i)$. $\alpha_t(i)$ is defined as the probability of the partial observation $o_1 o_2 \cdots o_t$ and state-i at time t, given the model λ. And $\beta_t(i)$ is defined as the probability of the partial observation sequence from time $t+1$ to the end, given state-i at time t and the model λ. Further, $\gamma_t(i, j)$ is defined as the probability to make transition from state-i to state-j at time t, given the observation sequence.

[Baum–Welch Algorithm]: Estimate parameters according to following equations repeatedly.

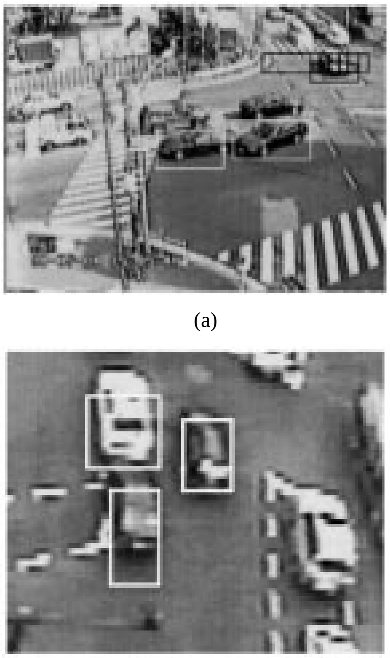
1) Initialize a_{ij}, $b_{ij}(k)$.
2) Estimate $\gamma_t(i, j)$ as follows:

$$\begin{aligned}\gamma_t(i, j) &= \frac{\alpha_{t-1}(i) a_{ij} b_{ij}(o_t) \beta_t(j)}{\sum_i \alpha_t(i) \beta_t(i)} \\ \alpha_t(i) &= P(o_1 o_2 \cdots o_t, q_t = S_i | \lambda) \\ \beta_t(i) &= P(o_{t+1} o_{t+2} \cdots o_T, q_t = S_i | \lambda). \end{aligned} \tag{16}$$

3) Estimate a_{ij}, $b_{ij}(k)$ as follows:

$$a_{ij} = \frac{\sum_t \gamma_t(i, j)}{\sum_t \sum_j \gamma_t(i, j)} \tag{17}$$

$$b_{ij}(k) = \frac{\sum_{t:\, o_t = k} \gamma_t(i, j)}{\sum_t \gamma_t(i, j)}. \tag{18}$$


(a)

(b)

Fig. 15. Bumping accidents. (a) Yokohama: Harajuku (courtesy of ASHRA Japan). (b) Tokyo; Surugadai.

For recognition, Trellis variables are calculated inductively as follows.

[Trellis Calculation]:

1) Initialization:

$$\alpha_1(j) = a_{0j} b_{0j}(o_1), \qquad j = 0, 1. \tag{19}$$

2) Induction:

$$\alpha_{t+1}(j) = \left(\sum_{i=1}^{N} \alpha_t(i) \alpha_{ij} \right) b_{ij}(o_{t+1}), \qquad j = 0, 1. \tag{20}$$

3) Termination:

$$P(o_1 o_2 \cdots o_T | \lambda) = \sum_{j=1}^{N} \alpha_T(j) \tag{21}$$

The model λ is provided with each category to be recognized. And the model which has the most likely probability with a test sequence is determined.

D. Experimental Results

We consider the following three situations in this experiment. These three situations resemble each other and it is interesting to investigate whether our HMM algorithm can classify them in correct categories.

[Situation 1] **Bumping Accident** (Fig. 15).

[Situation 2] **Stop and Start in Tandem**. This case resembles bumping accident, but there is no impact such as in bumping. Also, the distance between the two objects is larger than in the case of bumping.

[Situation 3] **Passing**. One object passes alongside another stopped object, or moving objects pass each other.

Typical observation sequences for the three situations are as follows. Each observation sequence consists of 20 observation

TABLE II
RESULTS OF ACCIDENT RECOGNITION

		Trained Models		
		λ_1	λ_2	λ_3
Observations for Test	O^{1-1}	$\mathbf{-13.7}$	-53.3	-98.2
	O^{1-2}	$\mathbf{-13.2}$	-28.4	-87.9
	O^2	-24.1	$\mathbf{-15.1}$	-51.5
	O^2	-37.8	-126.0	$\mathbf{-9.5}$

numbers; each of those observation numbers corresponds to an image frame in which images are obtained at a rate of 10 frames/s

[Situation 1] O^{1-1} [Fig. 15(a)]
000113330008777000000
[Situation 1] O^{1-2} [Fig. 15(b)]
000012310007777770000
[Situation 2] O^2
000001221107870000000
[Situation 3] O^3
$001123333333322211000.$

An HMM is trained for each of the three situations, respectively, and parameters λ_1, λ_2, λ_3 can be obtained. Training HMM requires a great deal of data. In order to bootstrap our real date on the accident scene, we add observation sequences, adding small disturbances around them. Each HMM is trained with 40 sample observation sequences; some of which are manually generated.

O^{1-1} and O^{1-2} are observation sequences for real accidents that occurred at different intersections. Table II shows classification results of O^{1-1}, O^{1-2}, O^2, O^3 with the trained parameters λ_1, λ_2, λ_3. As the results are represented by $P(O^i|\lambda_j)$, all values are negative and a smaller absolute value means a better matching result. As shown in Table II, accidents were successfully detected by HMM. Although O^{1-1} and O^{1-2} are observation sequences of different accidents at different intersections, HMM successfully detected both. Other situations such as moving in tandem and passing were also discerned successfully.

V. CONCLUSIONS

Generally, it is difficult to track multiple vehicles without confusing them. In particular, tracking is very difficult at intersections where various kinds of occlusion and cluttered situations occur. In order to achieve robust tracking in occluded and cluttered situations, we have derived an algorithm, which we refer to as the Spatio-Temporal Markov Random Field Model, and evaluated it on real traffic images. We can successfully demonstrate the ability to track multiple vehicles at intersections with occlusion and clutter effects at the success rate of 93%–96%. Although the algorithm achieves such reliable tracking, it requires only gray-scale images; it does not assume any physical models, such as shape or texture, of vehicles.

By using such a reliable tracking method, it becomes possible to monitor and analyze traffic events at intersections in detail. Although this algorithm for accident detection has been demonstrated only on a small number of cases, due to the limitation of observed accidents at these intersections during our observation period—three cases during one-year observation period—its performance is excellent; we can confidently predict its promise. For our future work, we will collect more accident data to evaluate the method's effectiveness. We also plan to analyze other activities such as traffic rule violations and other dangerous behavior.

REFERENCES

[1] N. Peterfreund, "Robust tracking of position and velocity with kalman snakes," *IEEE Trans. Pattern Anal. Machine Intell.* , vol. 21, pp. 564–569, June 1999.

[2] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. J.Computer Vision*, vol. 1, pp. 321–331, 1988.

[3] S. M. Smith and J. M. Brady, "ASSET-2: Real-time motion segmentation and shape tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, pp. 814–820, Aug. 1995.

[4] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. CVPR 1999*, June 1999, pp. 246–252.

[5] H. Leuck and H.-H. Nagel, "Automatic differentiation facilitates OF-integration into steering-angle-based road vehicle tracking," in *Proc. Conf. Conputer Vision and Pattern Recognition (CVPR) '99*, 1999, pp. 360–365.

[6] W. F. Gardner and D. T. Lawton, "Interactive model-based vehicle tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, pp. 1115–1121, Nov. 1996.

[7] S. Kamijo, Y. Matsushita, K. Ikeuchi, and M. Sakauchi, "Traffic monitoring and accident detection at intersections," in *Proc. IEEE ITSC'99*, Oct. 1999, pp. 703–708.

[8] R. Gangisetty, "Advanced traffic management system on I-476 in Pennsylvania," in *Proc. IEEE ITSConf '97*.

[9] M. Wener and W. von Seelen, "Using order statistics for object tracking," in *Proc. IEEE ITS Conf '97*.

[10] P. H. Batavia, D. A. Pomerleau, and C. E. Thrope, "Overtaking vehicle detection using implicit optical flow," in *Proc. IEEE ITS Conf '97*.

[11] J. C. Rojas and J. D. Crisman, "Vehicle detection in color images," in *Proc. IEEE ITS Conf '97*.

[12] N. Zeng and J. D. Crisman, "Vehicle matiching using color," in *Proc. IEEE ITS Conf '97*.

[13] A. H. S. Lai and N. H. C. Yung, "A video-based system methodology for detecting red light runners," in *Proc. IAPR Workshop on MVA '98*, pp. 23–26.

[14] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecci, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671–680, 1983.

[15] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distribution, and the bayesian restoration of images," *IEEE Trans. pattern Anal. Machine Intell.*, vol. PAMI-6, pp. 721–741, June 1984.

[16] R. Chellappa, S. Chatterjee, and R. Bargdzian, "Texture synthesis and compression using Gaussian–Markov random field models," *IEEE Trans. Syst. Man Cybern.*, vol. 15, Feb. 1985.

[17] P. Andrey and P. Tarroux, "Unsupervised segmentation of Markov radom field modeled textured images using selectionist relaxation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 20, Mar. 1998.

[18] V. Kettnaker and M. Brand, "Minimum-entropy models of scene activity," in *Proc. CVPR*, 1999, pp. 281–286.

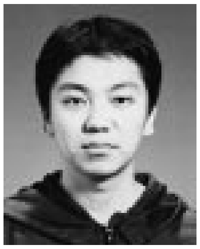
[19] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden Markov model," in *Proc. CVPR*, 1992, pp. 379–385.

[20] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, pp. 257–286, Feb. 1989.

[21] S. E. Levinson, L. R. Rabiner, and M. M. Sondhi, "An introduction to the application of the theory of probabilistic functions of a Morkov process to automatic speech recognition," *Bell Syst. Tech. J.*, vol. 62, no. 4, pp. 1035–1074, 1983.

Shunsuke Kamijo received the B.S. and M.S. degrees in physics from the University of Tokyo, Japan, in 1990 and 1992, respectively. He studied X-ray astrophysics for the M.S. degree.

Since 1998, he has been working toward the Ph.D. degree at the Institute of Industrial Science, University of Tokyo. His major is information engineering, and his recent interests include computer vision and stochastic models. Currently, he is now interested in applying computer vision techniques and stochastic models to image analyses of traffic activities.



Yasuyuki Matsushita received the B.E. and M.E. degrees in electrical engineering from the University of Tokyo, Japan, in 1998 and 2000, respectively.

He has been working toward the Ph.D. degree in electrical engineering at the Institute of Industrial Science, University of Tokyo since 2000. He is mainly interested in 3-D modeling of traffic images.

Katsushi Ikeuchi (M'78–SM'95–F'98) received the B.Eng. degree in mechanical engineering from Kyoto University, Kyoto, Japan, in 1973 and the Ph.D. degree in information engineering from the University of Tokyo, Japan, in 1978.

After working at the Artificial Intelligence Laboratory at Massachusetts Institute of Technology, Cambridge, the Electrotechnical Laboratory of the Ministry of International Trade and Industries, and the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, he joined the University of Tokyo, in 1996, where he is now a Professor at the Institute of Industrial Science.

Dr. Ikeuchi received various research awards, including the David Marr Prize in computational vision in 1990 and the IEEE R&A K-S Fu Memorial Best Transactions Paper Award in 1998. In addition, in 1992, his paper, "Numerical Shape from Shading and Occluding Boundaries," was selected as one of the most influential papers to have appeared in the *Artificial Intelligence Journal* within the past 10 years.

Masao Sakauchi received the B.S. degree in electrical engineering from the University of Tokyo, Japan, in 1969 and the M.S. and Ph.D. degrees in electronics engineering from the University of Tokyo in 1971 and 1975, respectively.

He is the Director General and a Professor at the Institute of Industrial Science, University of Tokyo. He is also the Project Leader of two big research projects on Multimedia Mediation Systems and ITS. He has written more than 260 refereed papers in the research fields for multimedia databases, multimedia systems, image processing and understanding, spatial data structures, and geographical information systems and fault-tolerant computing.

Dr. Sakauchi has acted as General Chairman of four international conferences and workshops, including the IEEE International Workshop on Machine Vision and Machine Intelligence (1987), IAPR and IEEE International Conference on Document Analysis and Recognition (ICDAR'93) (1993), International Symposium on Multimedia Mediation Systems (2000). He was Program Chairman of three international conferences, including the IEEE International Conference of Multimedia Processing and Systems (IEEE Multimedia '96) (1996), ITSC '99 (1999), and was Organizing and Program committee member of many international conferences. He also served as Chairman of the Technical Committee on Machine Vision and Machine Intelligence, IEEE IE society (1985–1992), was Associate Editor of the IEEE MULTIMEDIA MAGAZINE, (1993–1999) IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS. From 1984 to 1992, he was Editor in Chief of *Transactions on Information and Systems* of the Institute of Electronics, Information and Communication Engineers (IEICE) in Japan. From 1989 to 1991, he was Chairman of three technical committees on Image Engineering of IEICE of Japan, and the Institute of Television Engineers of Japan from 1987 to 1993.