自 动 化 学 报
ACTA AUTOMATICA SINICA

3-D Model Based Visual Traffic Surveillance[1)]

LOU Jian-Guang LIU Qi-Feng TAN Tie-Niu HU Wei-Ming
(*National Laboratory of Pattern Recognition, Institute of Automation,*
Chinese Academy of Sciences, Beijing 100080, *P. R. China*)
(E-mail: {jglou, qfliu, tnt, wmhu}@nlpr.ia.ac.cn)

Abstract Visual surveillance in dynamic scenes is an active research topics in computer vision. The aim of visual surveillance is to make it possible that the computer can watch or monitor a scene by automatic localization, tracking and recognition of moving objects and semantic interpretation of their behaviors in the watched scene. This paper aims to realize a task-specific traffic surveillance system which consists of modules for camera calibration, model visualization, pose refinement, tracking, trajectory-based semantic interpretation of vehicle's behaviors, etc. In this paper, we describe each module to give readers a comprehensive view of a visual surveillance system, and also discuss possible further work.

Key words Visual surveillance, pose refinement, wire-frame model, semantic interpretation, tracking filter

1 Introduction

Visual traffic surveillance tries to make it possible that the computer can automatically localize, recognize and track the moving vehicles in image sequences by the analysis of the image sequences captured by cameras from wide-area, real-world scenes in natural conditions. Furthermore, the computer can interpret the motions and behaviors of the tracked objects based on the tracking results, and finally, give semantic descriptions. All of these will be very helpful for not only regular traffic management, but also instant response when abnormal situation happens in the watched scene. In the last two decades, visual surveillance has attracted much interest in the area of computer vision because of its tremendous application prospect[1~3].

However, it remains elusive to build a general-purpose computer vision system which can work under a variety of scenes. Experience suggests that it should be possible to build a task specific system by exploiting task-based *a priori* knowledge. In this paper, we discuss visual surveillance in traffic scenes where specific knowledge is exploited, including three-dimensional geometric descriptions of individual vehicles, vehicle motion models, and some reasonable assumptions or constraints, such as the ground-plane constraint (GPC)[4], the weak perspective projection assumption[5], *et al*. All of these can provide explicit guidance for localization, recognition and tracking, and can significantly simplify the problem and reduce the computational complexity.

In most traffic scenarios, the target objects are known and three-dimensional geometric descriptions for these objects can be established in advance through measurement, CAD modeling or computer vision techniques (i. e., structure from motion[6]).

In conventional approaches, the image is first analyzed to extract global image features (such as straight lines), and then they try to establish the correspondence between the image features and the model features[7]. It is far from trivial because a search through all possible correspondences among image and model features is unpractical, especially when a large number of irrelevant image features are present[4]. However, top-down ap-

1) Supported by NFSC(59825105)
Received October 20, 2002; in revised form January 16, 2003
收稿日期 2002-10-20; 收修改稿日期 2003-01-16

proaches with hypothesis driven scheme can successfully avoid the feature correspondence problem, as the matching can be implicitly determined as a byproduct of the hypothesis[8].

In the past decade, we have developed a 3-D model-based visual traffic surveillance system[4,5,9~14] which is based on hypothesis driven vehicle tracking algorithms. In this paper, we present a brief introduction of the system and some recent new steps on it.

System overview

The work described in this paper is a vision-based vehicle tracking system for automatic identification and description of the behaviours of vehicles within traffic scenes. A schematic diagram of the system is shown in Fig. 1. We assume that the camera is static and calibrated. We also assume that 3-D wireframe models of vehicles have already been established as part of the task specific knowledge. In our system, image sequences captured from a CCTV camera are first input into the motion detection module to identify image regions where significant motion occurs. These regions are called regions of interest (ROI) because they are likely to contain road vehicles. For each detected ROI in a specific frame, either the predictive tracking module or the pose initialization module is activated according to whether it is occurring for the first time. An initial pose for the vehicle in the ROI is generated in both cases, which is further refined by the pose refinement module to deliver the final tracking result. The tracking results are the start point of our high-level sub-system which will give semantic descriptions of the behaviours of vehicles. The geometrical results provided by the low-level tracking sub-system are first converted to motion concepts, and then different concept patterns are recognized as action by action model and high-level reasoning. In our system, the final aim is to obtain natural language descriptions for a surveillance scene. Some of the recognized actions are selected to output, and the system generates natural language sentences by some grammar rules.

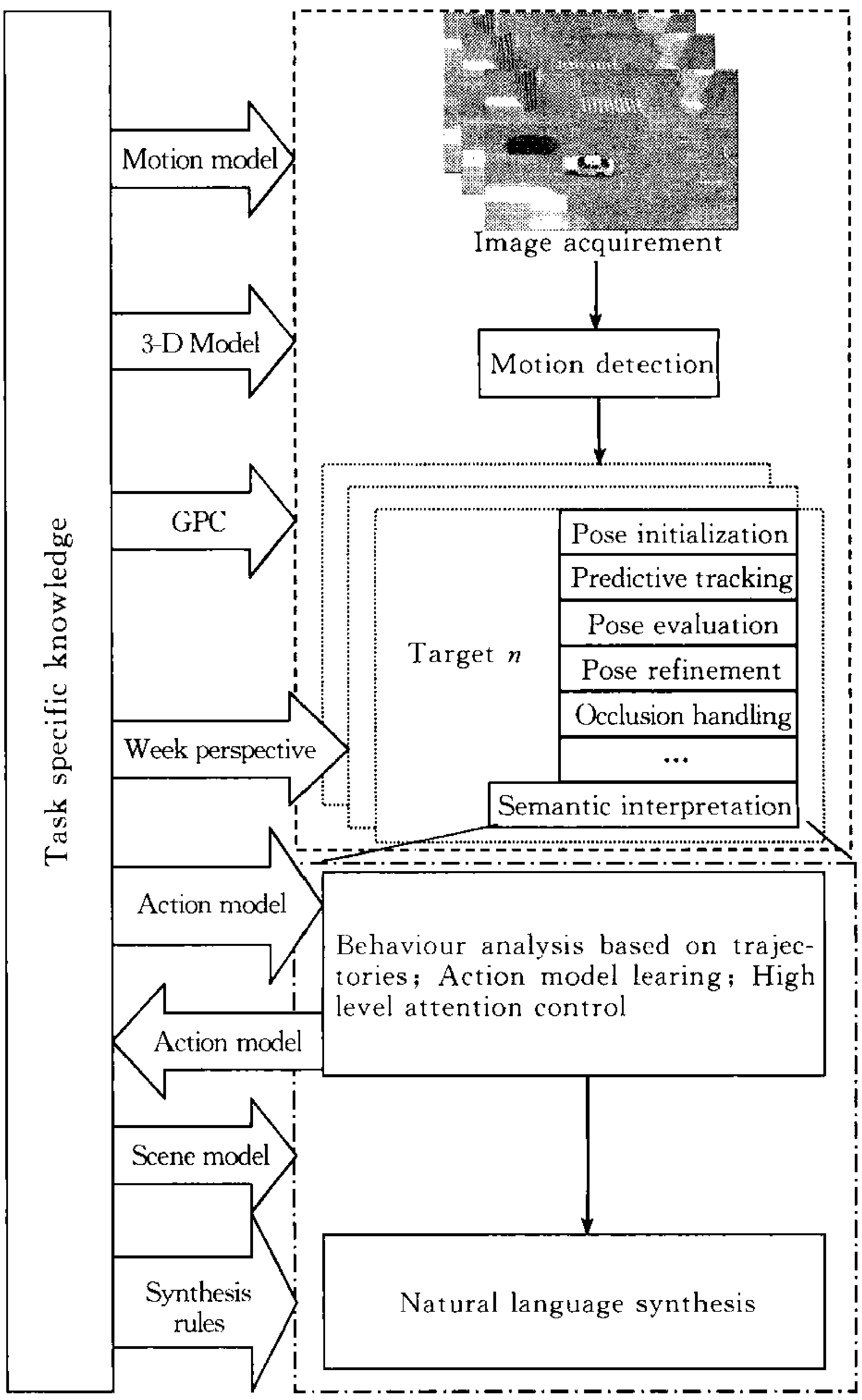


Fig. 1 System diagram

This paper concentrates on some of the new steps and ideas which are recently presented in our surveillance system, though advances in other parts of the system have also been made. In Section 2, we discuss the illumination invariant motion detection algorithm adopted in the system, where a Gaussian process is used to model every pixel on the background image. Section 3 presents a camera calibration method which is very convenient for traffic scene. Based on the calibration result provided by the method, we improve our model visualization module. In Section 4, we discuss and compare two pose evaluation functions. An improved EKF with a vehicle motion model is proposed for predictive tracking, which can reduce the filter′s sensitivity to the model uncertainty. Section 6 talks about a simple trajectory based behaviour interpretation algorithm. Some experimental results are given in sections where the relative algorithms are discussed. We also present overall results of the whole system in Section 7. In Section 8, we draw a conclusion and list the future work. Further details of the system may be found in our previous work[4,5,9~14].

2 Motion detection

Robust and efficient motion detection is an important preprocessing step to solve many problems in the area of computer vision including visual surveillance. One of the widely used approaches to this problem is background subtraction which assumes that the background image describes the stationary portion of the scene, and moving objects can be identified as those regions of pixels in the image that differ significantly from the background.

In the past twenty years, color or intensity based approaches[15,16] and range based approaches[17,18] are proposed for change detection. In [19], the authors integrate the range and color cues into change detection. Because range based algorithm is limited to multiview image sequences as the range information is usually obtained by stereo vision, color or grayscale based approaches are more suitable for monocular vision applications. The algorithm presented in this paper is precisely of this kind.

To establish a change detection framework that is flexible to deal with variations in lighting and the presence of moving shadows, we first use illumination invariant characteristics to describe the model of background. After this, the connectivity information is integrated into the background-foreground classification operation using Bayesian rules.

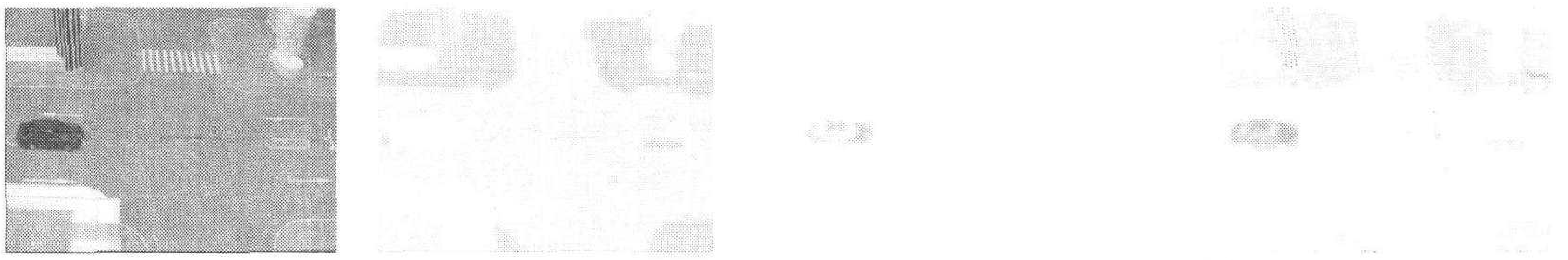
The basic steps of background subtraction algorithm include background modeling and pixel classification.

2.1 Background modeling

In our approach, we combine color cues and brightness information to construct a statistic background model. The color components in our approach are preprocessed by homomorphic filtering to avoid the influence of lighting changes. Generally, the scene illumination varies smoothly over space and locates at low frequency part in frequency domain. In addition, reflection components locate at relatively high frequency part. According to this, we can separate them by homomorphic filtering.

In Fig. 2, we can find that significant difference can be seen between the background and foreground in the reflection images.

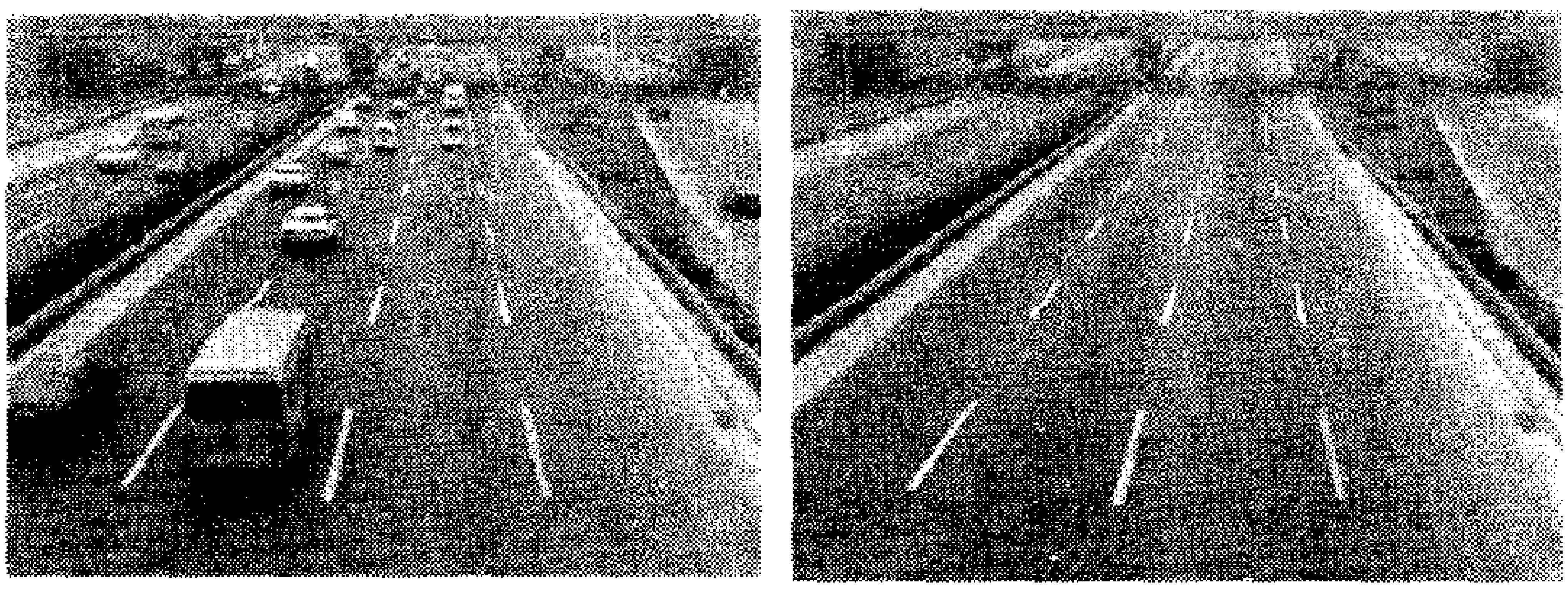
Assume that the pixel process belonging to the background is a set of Gaussian processes. Every background image pixel is modeled by a 4-tuple $\langle E_R, \sigma_R, l, \sigma_l \rangle$, where E_R denotes the expected illumination invariant color vector [Err Erg Erb], $\sigma_R = [\sigma_r \quad \sigma_g \quad \sigma_b]$, l is the brightness component and σ_l denotes the standard deviation of l.



(a)the original image (b)the red reflection component (c)the green reflection component (d)the blue reflection component

Fig. 2 Color reflection components

To estimate these parameters, the method which has been proposed in our previous work[20] is employed. For example, Fig. 3 shows an estimated background image for the left highway scene.



(a) (b)

Fig. 3 Background acquirement

2.2 Pixel classification

In our approach, the connectivity information is integrated into this step, because most moving objects always manifest themselves as compact connected regions.

First, we use a simple threshold operation: a pixel in the current image is 'foreground', if the pixel has significant difference of reflection characteristics from the values in the reference image; it is selected as 'background', if the pixel has similar reflection components to the same pixel in the background image. After this step, pixels in the current image have been roughly divided into 'background' and 'foreground'.

Then, we automatically select a threshold for every pixel in current image by integrating the connectivity constraint into classification. The main idea is based on the perception that the possibility of a pixel belonging to 'foreground' will increase if the number of its neighbors belonging to 'foreground' increases. The detailed information can be found in [9].

3 Camera calibration and model visualization

In our 3-D model-based tracking system, camera parameters are necessary for the 3-D model to project onto the image plane. Although camera calibration is a classic problem in computer vision and there have been many methods for this problem, but, almost all of them are not very convenient to use in traffic scenes. We try to use a simple but convenient calibration tool. In our system, we first obtain the homography H between the ground plane and the image plane by some corresponding points between these two planes. Then, we use the height of the camera (h) and some lines which are vertical to the ground plane to calibrate the camera.

For a pin-hole model, the project matrix is $M=A\cdot\begin{bmatrix}R & \boldsymbol{t}\\ \mathbf{0} & 1\end{bmatrix}$, furthermore the homography between the ground plane and the image plane should be

$$H = A[\boldsymbol{r}_1 \quad \boldsymbol{r}_2 \quad \boldsymbol{t}] = [\boldsymbol{h}_1 \quad \boldsymbol{h}_2 \quad \boldsymbol{h}_3] \tag{1}$$

where A is the camera's intrinsic parameter matrix, $\boldsymbol{r}_1$, $\boldsymbol{r}_2$ and $\boldsymbol{r}_3$ are three column vectors of rotation matrix R, and $\boldsymbol{t}$ is the translation parameter.

We can obtain the homography H up to scale if there are more than four pairs of corresponding points between the ground plane and the image plane.

Constraint 1

If the coordinate of optical center is (x_c, y_c, h), we have

$$x_c\boldsymbol{h}_1 + y_c\boldsymbol{h}_2 + \boldsymbol{K} + \boldsymbol{h}_3 = \mathbf{0} \tag{2}$$

where $\boldsymbol{K}=hA\boldsymbol{r}_3$.

Constraint 2

Given a line l^* which is vertical to the ground plane and its projection l on the image plane, we can find that line $H^{\mathrm{T}}l$ is on the ground plane and goes through the point $(x_c, y_c, 0)$.

According to these two constraints, given the height of camera and two vertical lines, we can calculate x_c, y_c and $\mathbf{K}$. Then, we can further calculate all the camera parameters.

In fact, during our model visualization process, it is not necessary to calculate individual parameters. We use the following steps to visualize the 3-D models as described in Fig. 4.

Step 1. Set (x_c, y_c, h) as the perspective center and project the models onto the ground plane. For any point in world coordinate system (x_w, y_w, z_w), the projection on the ground plane can be calculated as

$$\begin{bmatrix} x'_w \\ y'_w \end{bmatrix} = \begin{bmatrix} x_w \\ y_w \end{bmatrix} + \frac{z_w}{h - z_w}\left(\begin{bmatrix} x_w \\ y_w \end{bmatrix} - \begin{bmatrix} x_c \\ y_c \end{bmatrix}\right)$$

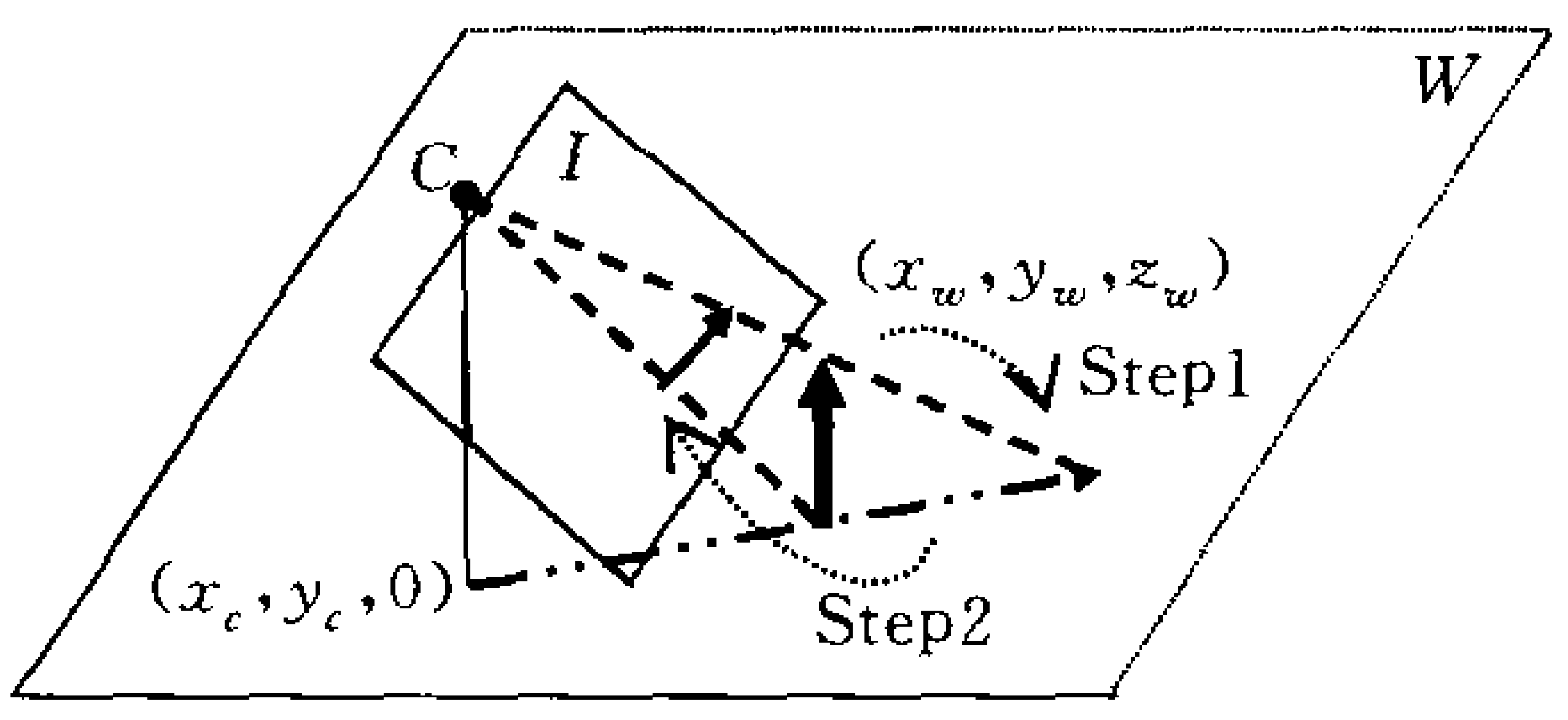


Fig. 4 Model visualization

We also should check the visibility of each model line at this step.

Step 2. Map the model's projection on the ground onto the image plane by the homography H. The mapping result is the final visualization of the model.

This method can avoid the additional error brought by the process of estimating individual camera parameters.

4 Vehicle localization and tracking

Vehicle localization and tracking is the basis of the visual traffic surveillance system, and it can heavily affect the performance of the system in terms of efficiency, accuracy, and robustness. The goal of vehicle tracking is to determine the dynamic states of the vehicles in every frame.

Tracking can be cast as an attempt to find the optimal estimation of the state in a discrete-time dynamic system. Let X_t be the state of the object and I_t the image at time t. Then tracking is to estimate the current state X_t, given the images from time 1 to time t $I_{1:t} = \{I_i, i = 1, 2, \cdots, t\}$. Note that X_t is a general concept that could contain various information of the object, e. g. position, translational or angular velocity, acceleration and shape.

From the Bayesian point of view, it is required to construct the a posteriori probability of X_t:

$$p(X_t \mid I_{1:t}) = p(I_t \mid X_t) p(X_t \mid I_{1:t-1}) / p(I_t \mid I_{1:t-1}) \tag{3}$$

As shown in Equation (3), there are two main problems to be resolved: 1) how to estimate the image likelihood $p(I_t \mid X_t)$, namely to evaluate the match quality between the state of the object and the current image data (in this paper, the state can be substituted with the pose and this problem can be called pose evaluation for short); 2) how to estimate $p(X_t \mid I_{1:t-1})$, namely to predict the current state according to the last ones. In fact, the probabilities in Equation (3) are hard to be directly computed but can be estimated by other metrics.

4.1 Pose evaluation

Methods for pose evaluation differ in the selected properties of the object and the image cues. There are mainly four classes for pose evaluation: feature-based, blob-based, active contour and shape model-based methods. The feature point based method try to find the correspondence between some primitives (e. g. points, line segments or corners) in the current and the former frames[21,22]. The blobs related with the moving objects are used by the blob based method, which has close relationship with the methods of background subtraction[23]. The active contour-based method focuses on the contour information of the object and establishes corresponding energy model[24].

As mentioned in Section 1, visual traffic surveillance is a specific problem and many useful constraints and a prior knowledge can be exploited to simplify the problem of object localization. One of the most important *a prior* knowledge is the skeleton information of the vehicle, which can be expressed by a 3-D wire-frame shape model. In this paper, we use a 3-D shape model-based method. We establish several 3-D wire-frame models of the vehicles off-line. For a given pose, we project the 3-D shape model onto the 2-D image plane and evaluate the matching between the projection and the local image data (see Fig. 5).

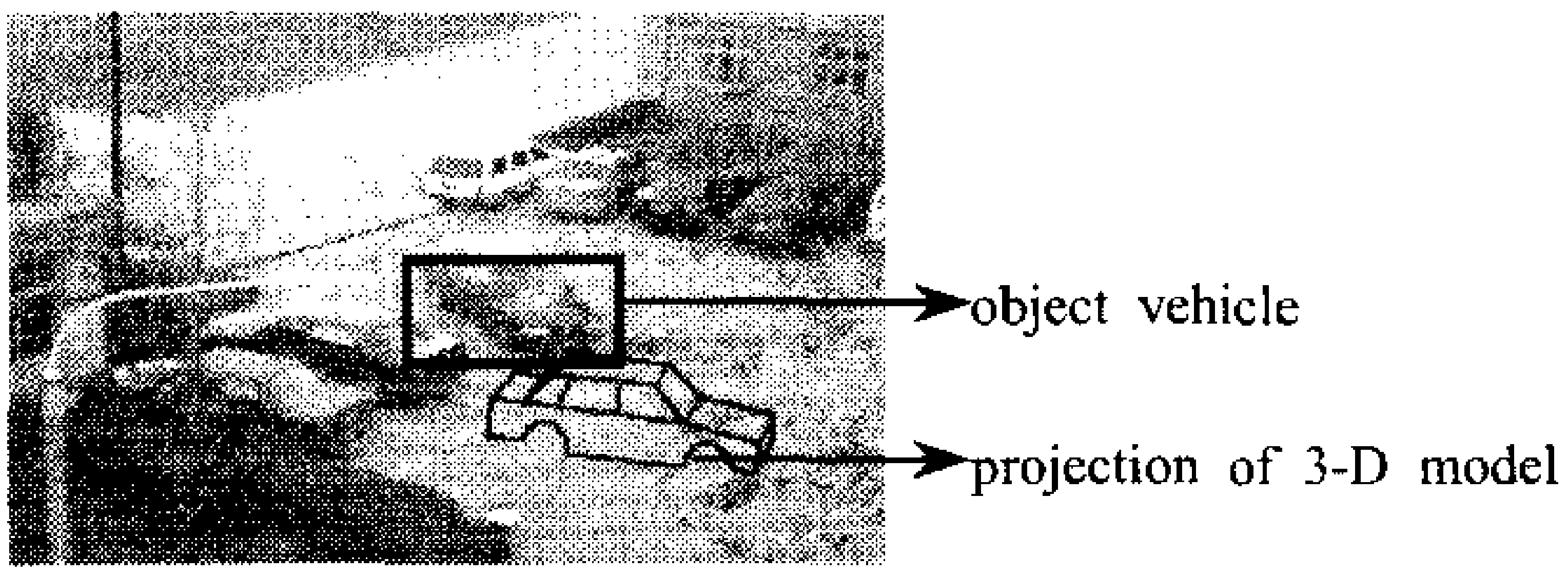


Fig. 5 Projection of 3-D model and the object vehicle

Generally, compared with 2-D methods, the 3-D model-based method can:

1) Utilize more a prior information to provide more accurate localization;

2) Directly and accurately determine the pose of the watched object in the 3-D world;

3) More robust against the changing of illumination, occlusion and clutter.

In model-based tracking, the key problem is pose evaluation. An intuitive idea is that first line segments in the image are extracted and then the given pose is evaluated by building up correspondences between these line segments and projection of the 3-D model[3,25]. However, this method is of high computational cost and sensitive to clutter. In this paper, we adopt a hypothesis driven strategy and directly use the image information (e. g. gradient and edge point) to avoid line segment exaction.

In our visual surveillance system, we adopt two methods for pose evaluation: 1) the Iconic method proposed by Brisdon[26], which is based on gradient magnitude; and 2) the PLS (Point to Line Distance, similar to Chamfer distance[27]) method proposed by us, which is based on a topological distance between the image edge points and the projected model line segments. In addition, Liu *et al.*[12] compare the two methods and find that the Iconic method is much faster and the curve of the PEF (pose evaluation function) of the PLS method is much smoother. For the sake of self-containedness, the following will outline the two methods.

a) Iconic method

The 3-D vehicle model is first instantiated at the given pose, and a set of visible line segments (L_1, L_2, $\cdots$, L_n) is obtained. As shown in Fig. 6, along a visible line segment $L_k(k=1,2,\cdots,n)$, several normals (N_1, N_2, $\cdots$, N_m) are taken at equal interval 2σ. For each normal, let δ be the interval of the sampled points on each normal, and $\varepsilon(v)=|I(v-\delta/2)-I(v+\delta/2)|$ the absolute value of the discrete derivative of image intensity, where $v\in\{\mu-8\delta,\cdots,\mu+8\delta\}$. Feature score e_k is computed as:

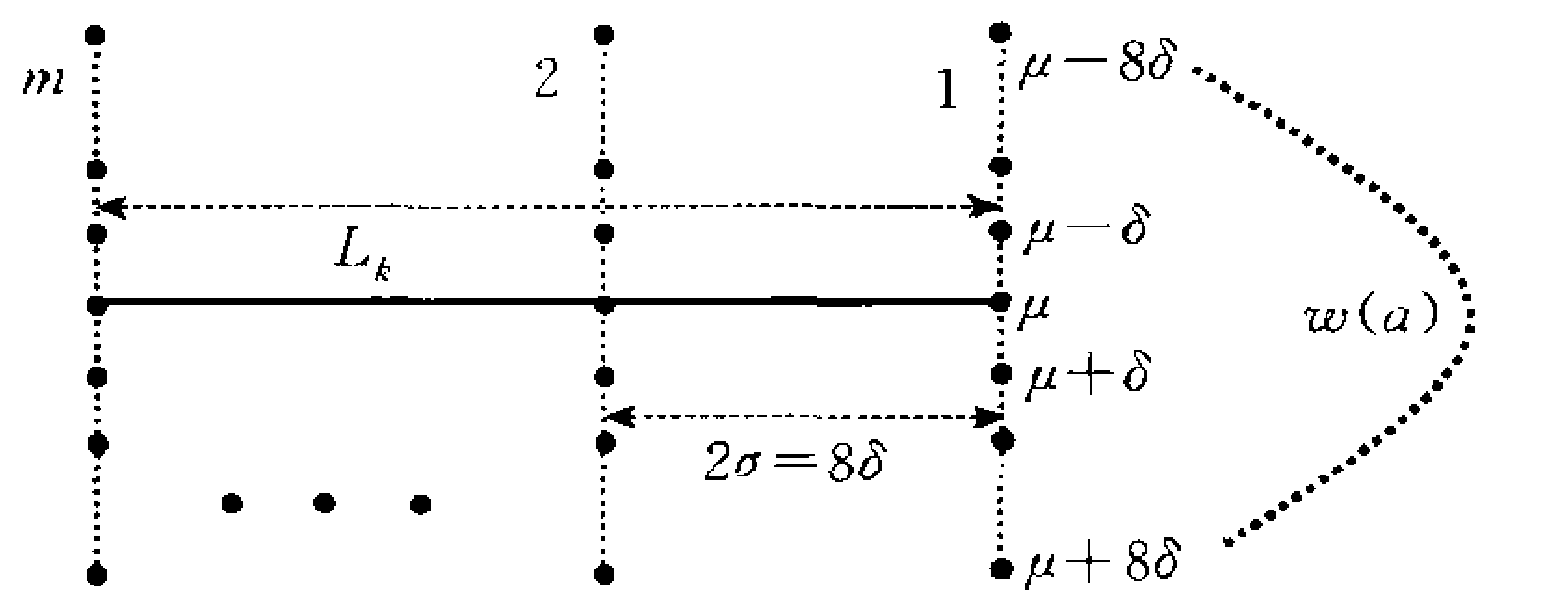


Fig. 6 Calculation of normal score and feature score

$$e_k=\frac{1}{m}\sum_{i=1}^{m}\sum_{v}\varepsilon(v)w\left(\frac{v-\mu}{\sigma}\right) \tag{4}$$

where $w(a)=\exp(-a^2/2)$ is a Gaussian window.

Under an assumption of a linear relationship[28], it is easy to convert e_k into the log-P-value:

$$\log P(e_k) = -\max[0, \beta_{L(k)} + \alpha_{L(k)} e_k] \tag{5}$$

where $P(e_k)$ is the probability that a line segment of the same length as $L(k)$ randomly placed on the image produces a feature score no less than e_k. Evaluation fuction is defined as:

$$E = -2\sum_{k=1}^{n} \log P(e_k) \tag{6}$$

Under the assumption of the visible line segments' independence, E has a χ^2 distribution with $2n$ degrees of freedom. The higher the score, the better the match (or the more accurate the given pose).

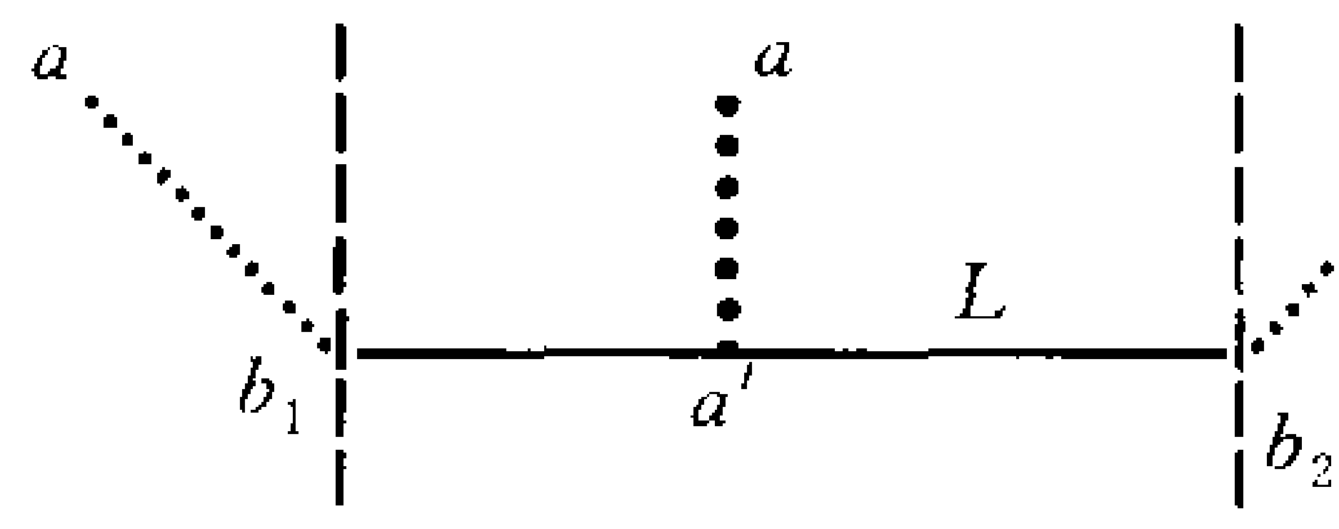


Fig. 7 PLS distance

b) PLS method

Given a point a and a line segment L specified by two end points b_1 and b_2, their PLS distance is defined as follows (see Fig. 7):

$$D(a,l) = \begin{cases} |\overrightarrow{aa'}|, & \text{if } \overrightarrow{ab_1} \cdot \overrightarrow{b_2b_1} > 0 \text{ and } \overrightarrow{ab_2} \cdot \overrightarrow{b_1b_2} > 0 \\ \min\limits_{b\in\{b_1,b_2\}} |\overrightarrow{ab}|, & \text{otherwise} \end{cases} \tag{7}$$

Given a set of edge points in a ROI (region of interest) $I=\{I_{i,j}\}$ and a set of visible line segments of the projected model $L=\{L_p \mid 1 \leqslant p \leqslant N\}$, pose evaluation function in PLS method is:

$$H(I,L) = \sum_{j,k} \min_{p}(W_{j,k,p} D(I_{j,k}, L_p))^2 \tag{8}$$

where $W_{j,k} = |R_p^{\mathrm{T}} Q_{j,k}|$ is the weight value ($Q_{j,k}$ is the unit gradient direction of $I_{j,k}$ and R_p the unit normal vector of the projected model line segment L_p).

Fig. 8 shows a real world scene for test (a) and the corresponding curves of PEFs (Pose Evaluation Function) of the two methods. From experimental results and our previous comparative work[12], we can learn that: 1) the computational cost of the Iconic method is much lower than that of the PLS; 2) the PEF surface of the PLS method is much smoother than that of the Iconic, which is helpful for refinement; 3) Iconic and PLS have similar abilities of localization; and 4) the peaks of both Iconic and PLS are not conspicuous enough under serious occlusion.

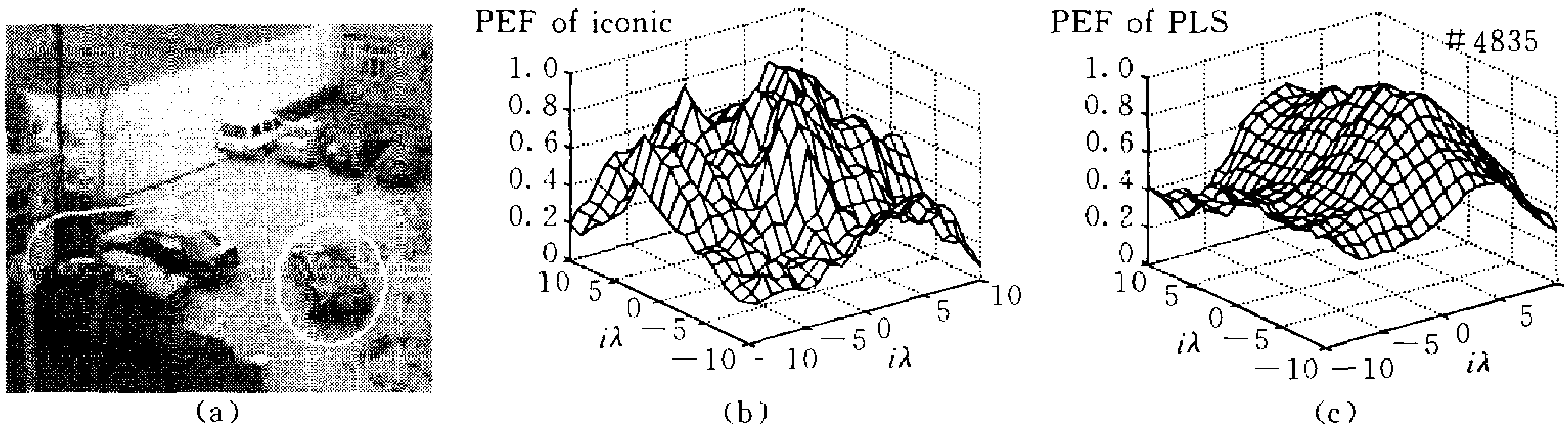


Fig. 8 (a) A real world scene image (the white ellipse indicates the object); (b) the curve of PEF of the Iconic method; (c) the curve of PEF of the PLS method

4.2 Pose initialization

The iterative process of pose refinement always starts from an initial pose. The initial pose is important because it could heavily affect pose refinement. During the tracking process, the initial pose can be provided by the prediction module of the tracking filter. For a new object, it is hard to estimate its initial pose. Sullivan *et al.* use the correspondence between the extracted line segments in the image and the 3-D model[1]. However establishing correspondence is always expensive computationally. We have improved this method[4] and reduced the computational cost at the same time.

4.3 Pose refinement

Given a PEF and the initial pose, pose refinement is a standard optimization problem. The adopted strategy for pose refinement is important and can strongly affect the performance of the whole system in terms of efficiency, accuracy and robustness.

In our system, because vehicles always move on the ground plane, GPC (Ground Plane Constraint) is applied here[5]. So the number of degrees of freedom (dof) of a vehicle is reduced from 6 to 3 and the pose P of the vehicle is comprised of translational parameters X, Y and one rotational parameter θ. For further reducing computational cost, WP (weak perspective assumption) is adopted here, which is valid in most traffic scenes. Under GPC and WP, we have demonstrated[5] that motion of vehicles can be decomposed into two independent motions both in the 3-D world and the image plane: translation and rotation. So we can use the Newton method to update the translation and rotation parameters alternatively to obtain a best pose.

4.4 Vehicle tracking

For an autonomous visual traffic surveillance system, the ability to track and predict the vehicle motion is important. First, because of the presence of noise and inaccuracies in image data and object models, the observed pose is often noisy. A filter is needed to obtain a smooth estimation of the tracked vehicle's motion parameters for semantic interpretation in high-level vision. Second, the predictive properties of the filter can be used to get an estimation of pose for the next frame based on the measurement of the preceding frames. An accurate prediction can simplify the measurement process and reduce the computational cost of searching in the object localization modules. In general, the accuracy of tracking and prediction depends on the structure of tracking filters that contain the dynamic model for the vehicle motion.

We model the vehicle motion as a fifth-order dynamic process with the state vector $\boldsymbol{X}=[x,y,v,\theta,\ \phi]^{\mathrm{T}}$, where $[x,y]^{\mathrm{T}}$ is the position of the vehicle on the ground plane, v is the velocity of the rear wheel, θ is the orientation of the whole vehicle, ϕ is the orientation of the front wheel. The dynamic equation and measurement equation can be described as follows which is deduced from a physical dynamic model shown in Fig. 9 (named bicycle model of vehicle motion):

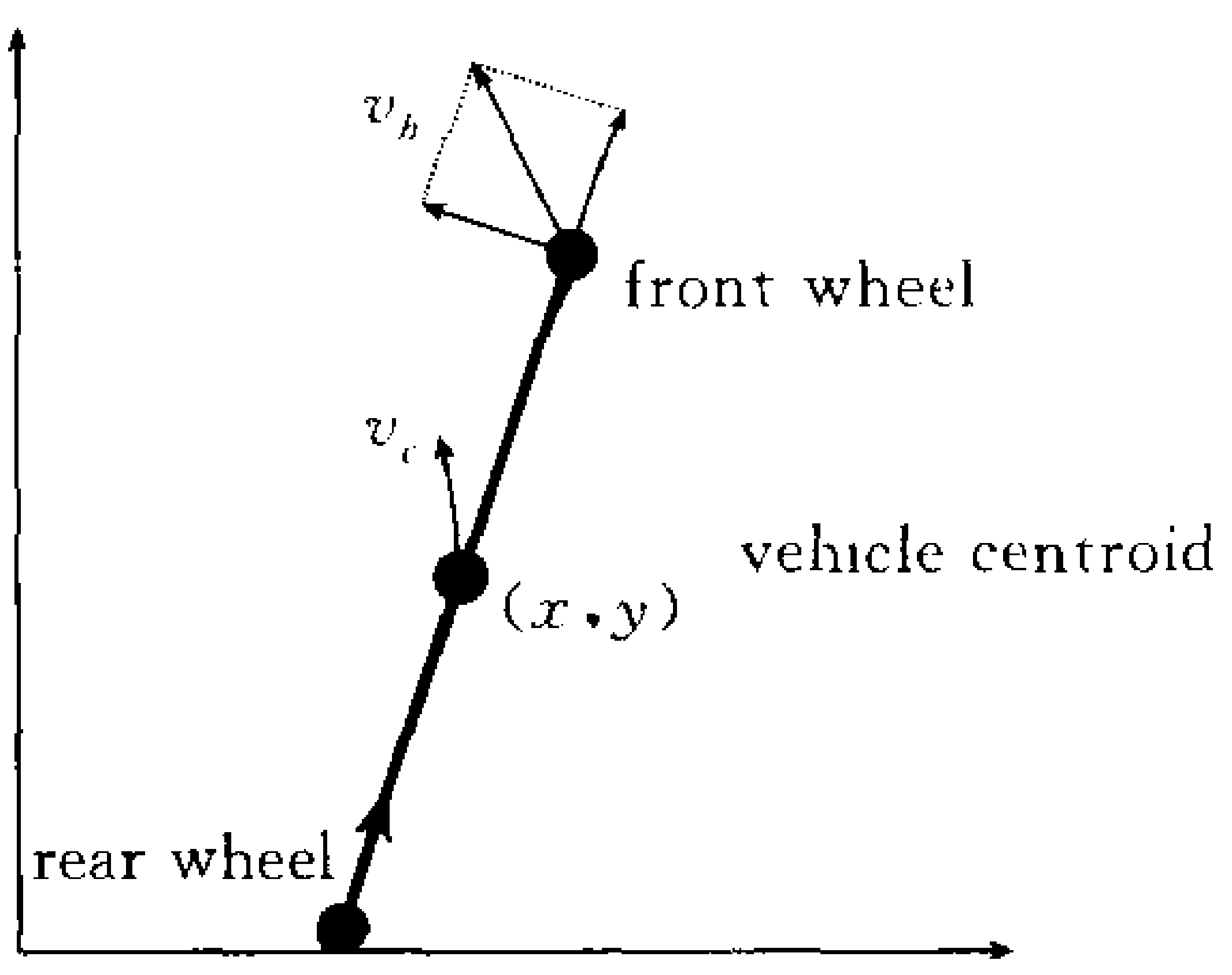


Fig. 9 The bicycle model of vehicle motion

$$\begin{cases} \dot{x} = \frac{v}{2\cos\phi}\sqrt{1+3\cos^2\phi}\cos\left[\theta + \operatorname{arctg}\left(\frac{\operatorname{tg}\phi}{2}\right)\right] \\ \dot{y} = \frac{v}{2\cos\phi}\sqrt{1+3\cos^2\phi}\sin\left[\theta + \operatorname{arctg}\left(\frac{\operatorname{tg}\phi}{2}\right)\right] \\ \dot{v} = a \\ \dot{\theta} = \frac{v \cdot \operatorname{tg}\phi}{l} \\ \dot{\phi} = b \end{cases} \tag{9}$$

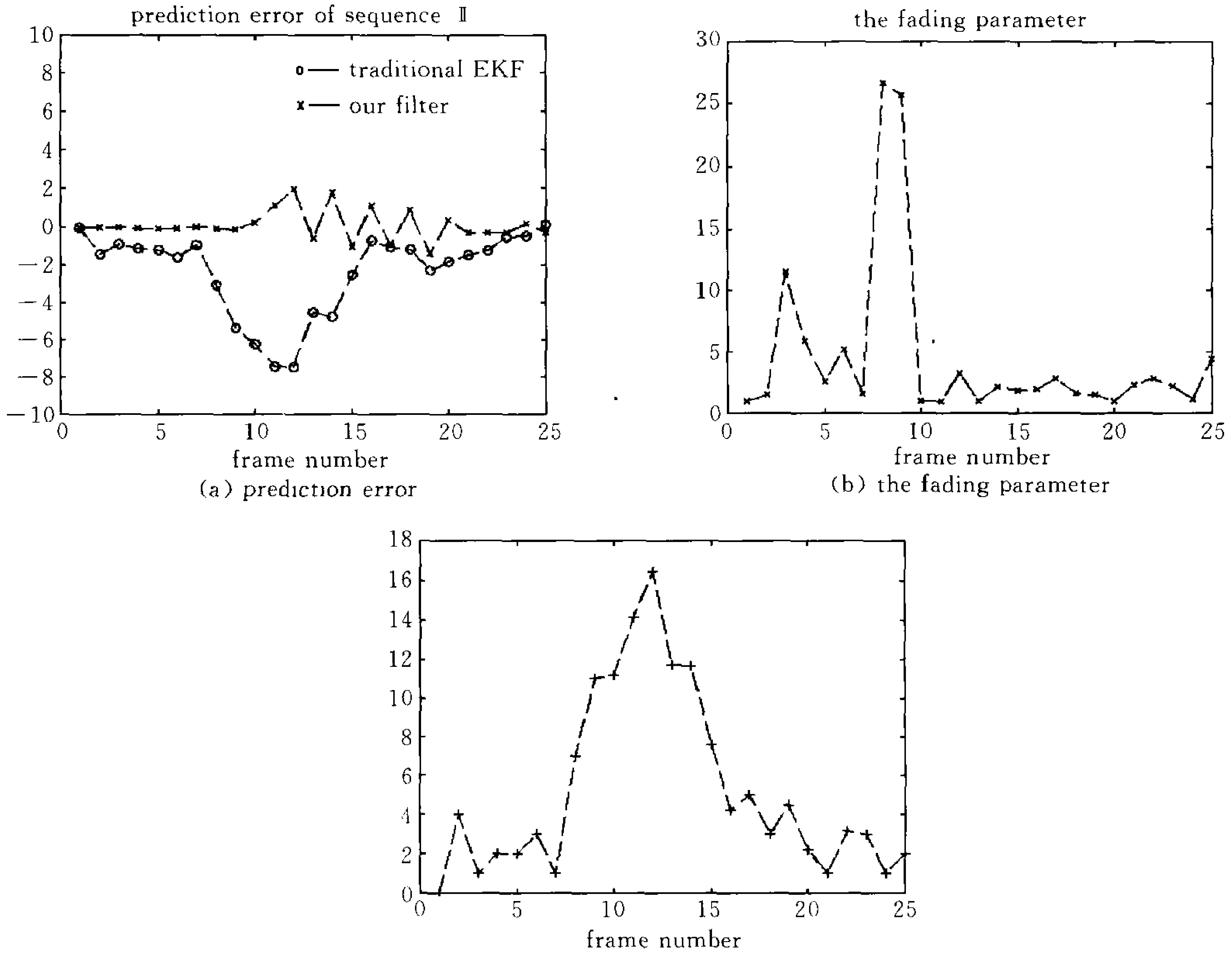
$$\boldsymbol{Y} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ v \\ \theta \\ \phi \end{bmatrix} + \boldsymbol{e} \tag{10}$$

In this dynamic model, a and b are used to describe the behavior of the driver. a reflects the driver's press on the accelerator or the brake, or changing the position of the gear; b reflects the turn of the steering wheel. Our dynamic model is more realistic and accurate than that of Koller *et al.*[3] and Maybank *et al.*[29].

There are two main classes of filter widely used in visual tracking problem: one is Kalman filter[2] and the other is Monte Carlo filter (e. g. CONDENSATION[3]). Kalman filter is faster than Monte Carlo filter, however, it require a very precise motion model which does not always hold because that the maneuver of a driver is very hard to model. It also assumes that all the noise is white noise. Monte Carlo filter does not hold this limit, however, its accuracy depends on the number of the sample particles, and thus it has high computation cost and not suitable for 3-D model based real time tracking.

In our system, in order to implement real time tracking, we adopt EKF (extended Kalman filter). In order to reduce the sensitivity of the filter to the model uncertainty, we modify the EKF by adding a new optimizing objective function. The idea is that once the model has changed, the residual error series would change immediately, and then we adapt the filter to the orthogonality condition (just like White Noise) in order that the filter's estimated states can track the system's real states quickly and accurately. If the model's parameters match the real system, the orthogonality condition will be self-satisfying for the EKF. But if the model changes over time, the traditional EKF's residual errors do not satisfy the orthogonality condition, and they reflect the instability of the model parameter. Because the measurement noise is assumed as White Noise, the residual error process should be White Noise. We adapt the filter to make sure that the residual error series has the similar characteristic with White Noise in order that the estimated states of the filter can track the system's states as quickly as the system's parameters change. This is achieved by using a fading parameter[13].

Experimental results show that our new filter is much more robust when the vehicle's motion behavior changes suddenly. In Fig. 10 (a), our filter has better performance than traditional EKF. From Figs. 10(b) and (c), we can find how the fading parameter works when dramatical changes of velocity made by the driver. However, it is still inevitable that our new filter also needs Gaussian noise assumption and its robustness against outliers and non-Gaussian noise is similar to that of common Kalman filter.



(a) prediction error

(b) the fading parameter

(c) the displacement of the car between two sequential frames over time (we use this to manifest the variation of car's speed)

Fig. 10　The result of sequence II

5 3-D wire-frame model

As shown in Fig. 11, we use 3-D wire-frame model to depict a saloon. This model can describe the skeleton properties of a vehicle well and provide valuable information for pose evaluation.

In addition, data structure storing the model in computer memory should be designed carefully to obtain a quick projection process. As shown in Fig. 12, we use facets as the

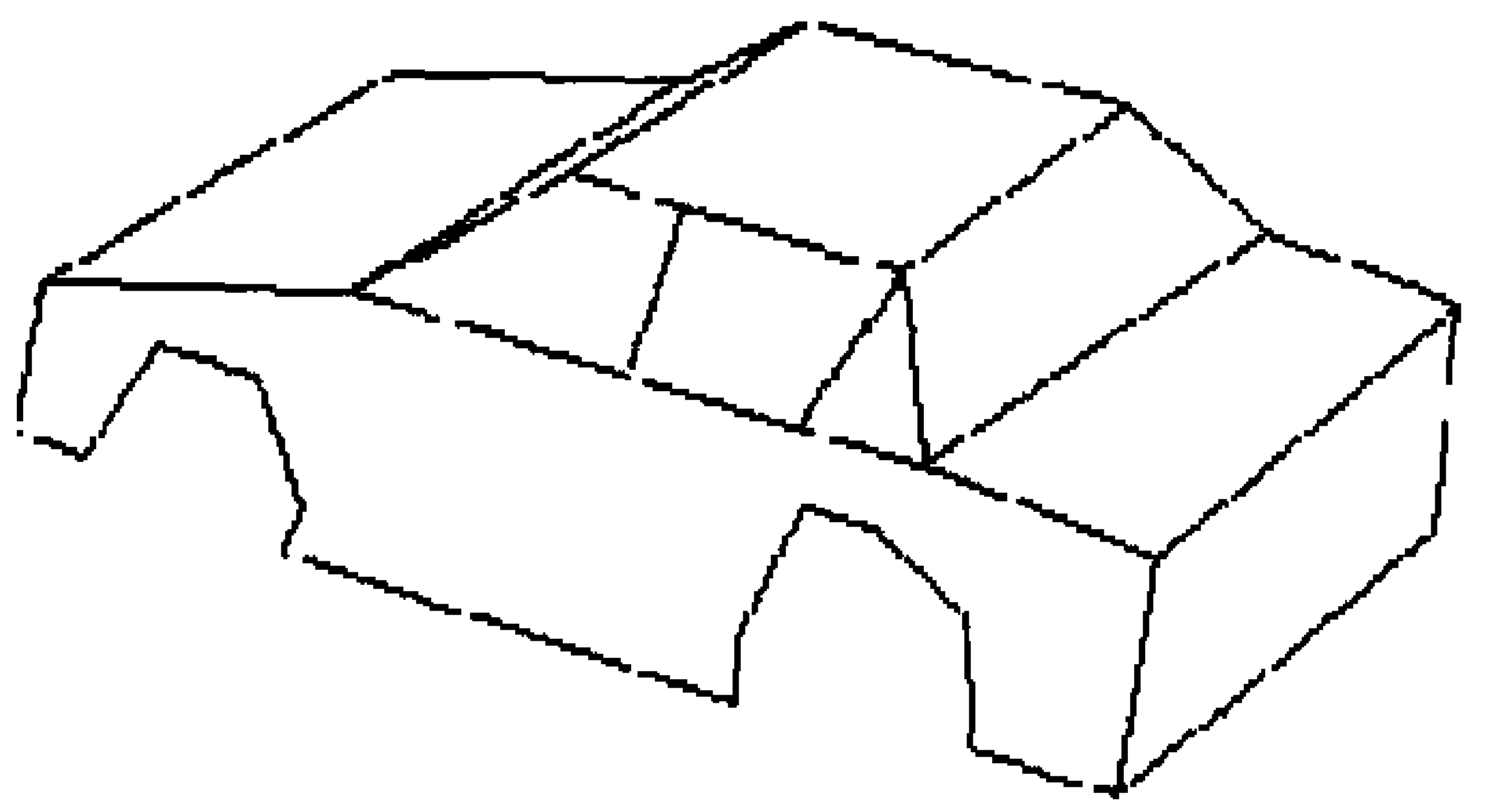
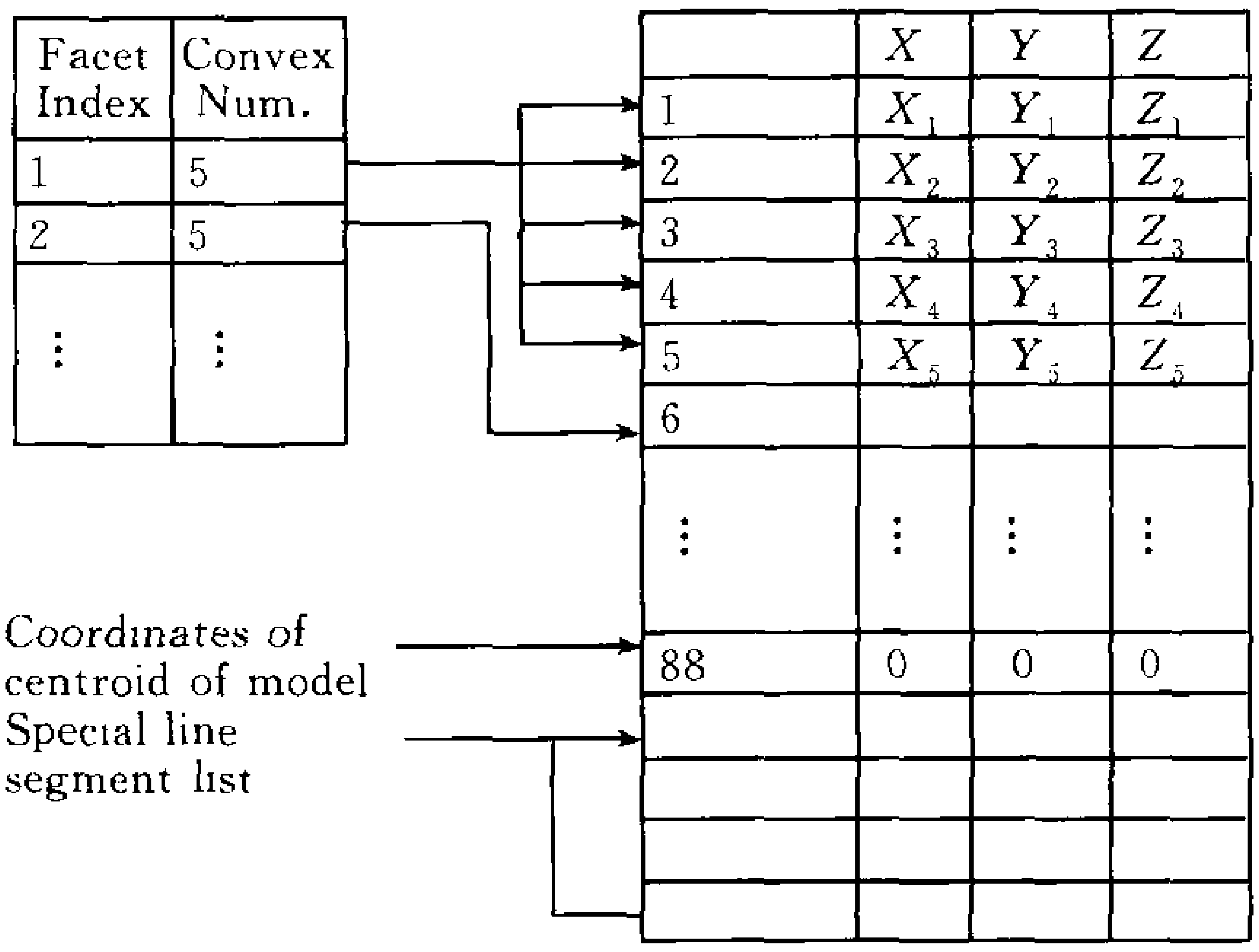


Fig. 11　3-D wire-frame model of a vehicle



Facet Index	Convex Num.
1	5
2	5
⋮	⋮

	X	Y	Z
1	X_1	Y_1	Z_1
2	X_2	Y_2	Z_2
3	X_3	Y_3	Z_3
4	X_4	Y_4	Z_4
5	X_5	Y_5	Z_5
6			
⋮	⋮	⋮	⋮
88	0	0	0

Fig. 12　Data structure of the 3-D model

main index elements and each facet consists of several vertexes. This strategy aims at decreasing computational cost, because visibility checker (the main component in projection process) is based on judging the relationship between two facets. The 3-D wire-frame model can be obtained by hand or SFM (structure from motion) method[6].

6 Semantic interpretation of vehicle's behaviour

Semantic interpretation plays a very important role in our surveillance system. An advanced visual surveillance system should be able to interpret what is happening in the dynamic scene, raise warning if some abnormal events occur, and also predict future actions of the tracked targets. In our visual surveillance system, low level tracking algorithms described above will provide the trajectories of moving targets in the watched scene.

6.1 Trajectory classification

Trajectory pattern analysis which can automatically classify the trajectories into several patterns is an important way for activity interpretation. We can often analyze the activities of the tracked target by analyzing the target's route, speed and other dynamic information contained in the target's trajectory. In our system, we designed a classification tree as illustrated in Fig. 13 with three layers. We use spatial information to cluster trajectories and then use dynamic information to classify the trajectories in every cluster into classes. Identical clustering algorithms are used in these operations.

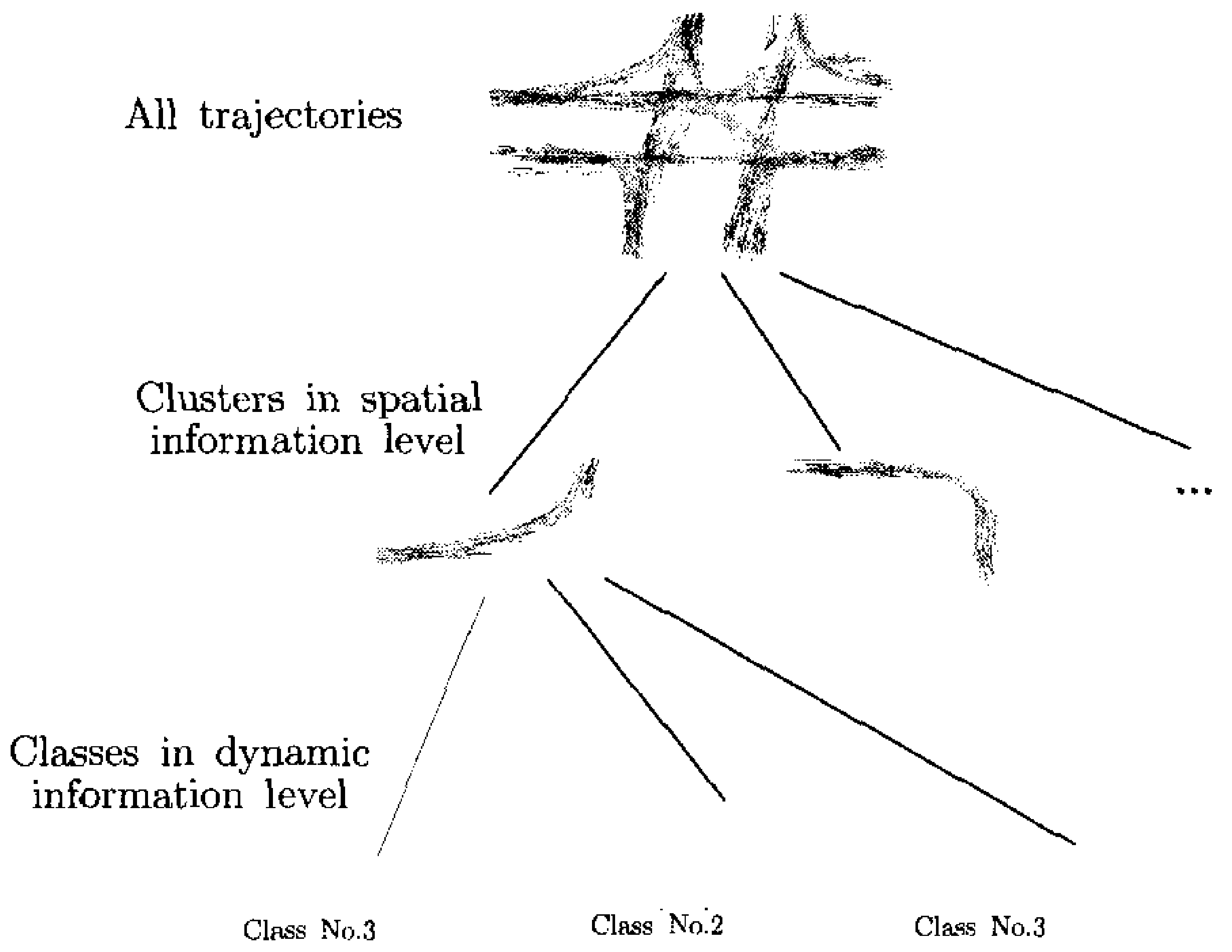

All trajectories
Clusters in spatial information level
...
Classes in dynamic information level
Class No.3
Class No.2
Class No.3

Fig. 13 Classification tree

How to measure similarity between two trajectories is the first problem we should tackle before we can analyze the trajectory's spatial and dynamic information. In [30], the authors utilize the percentage of overlapped pixels to measure similarity between trajectories, and an 80% overlap is assumed to identify the same trajectory class. It can be regarded as a global trajectory classification, but a simple threshold will sometime fail because of noise in the trajectory which is provided by a tracking system. After all, there are some open problems in visual tracking such as occlusion.

We define a distance formulation to measure the spatial similarity between trajectories which is similar to Hausdorff distance. This distance can be considered as global information of trajectories. Given two trajectories A and B, where A has t points and B has T points, their spatial distance D_c can be defined as:

$$D_c = \min_{A,B}\{D_{A,B}, D_{B,A}\} \tag{11a}$$

$$D_{A,B} = \max_{i=0,\cdots,t} \{ \min_{j=0,\cdots,T} (d_{i,j}) \} \tag{11b}$$

$$D_{B,A} = \max_{i=0,\cdots,T} \{ \min_{j=0,\cdots,t} (d_{i,j}) \} \tag{11c}$$

where $d_{i,j}$ is the Euclidean distance from the position of point i in one trajectory to point j in the other trajectory.

We also define a metric described below to measure similarity of trajectories' dynamic information.

$$D_t = \min_{A,B} \{ DV_{A,B}, DV_{B,A} \} \tag{12}$$

where

$$DV_{A,B} = \frac{\sum_{\text{all } i \text{ in } A} dv_{i,j}}{t+1}, \quad j = \arg\min_{0 \leqslant j \leqslant T} (d_{i,j})$$

$$DV_{B,A} = \frac{\sum_{\text{all } i \text{ in } B} dv_{i,j}}{T+1}, \quad j = \arg\min_{0 \leqslant j \leqslant t} (d_{i,j})$$

And $dv_{i,j}$ is the difference from the speed of point i in one trajectory to point j in the other trajectory.

Based on these similarity measurements, we use a C-Mean like clustering method to learn the activity patterns (See [14]).

In Fig. 14, we demonstrate our algorithm in a real world scene, and 6 classes of 14 learned classes are listed.

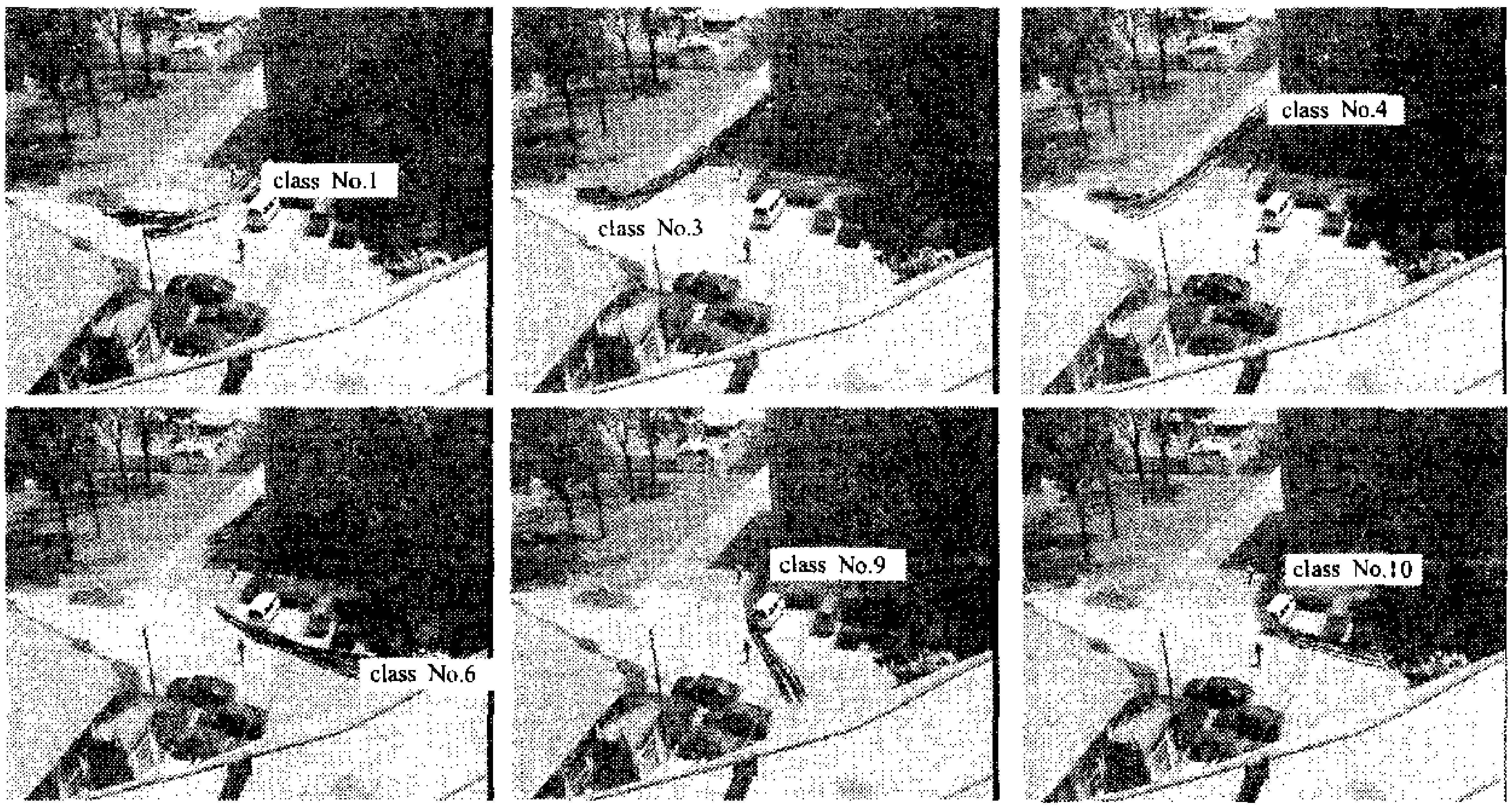


Fig. 14 Some trajectory classes

6.2 On-line classification

For every new input trajectory, Bayesian classifier is implemented to do the classification. Here, we denote the distance from one point in a trajectory A to the corresponding point in the representative trajectory as $d_i = \min_{1<j<T} (d_{i,j})$. In addition, we assume that these $d_i, i = 1, \cdots, t$, satisfy Gaussian distribution. This assumption is reasonable when the two trajectories belong to the same class. Furthermore, the joint probability density is

$$p(x \mid k) = \prod_{i=1}^{t} p(x_i \mid k) \tag{13}$$

where $p_i(x \mid \mathrm{k})$ is the Gaussian distribution density of point x in a trajectory which belongs to class k. The parameters of these Gaussian distributions can be estimated by calculating the scatter matrix in each cluster. The post probability density will be:

$$p(k \mid x)=\frac{P(k)\cdot p(x \mid k)}{\sum_{\text{all } k} P(k)\cdot p(x \mid k)} \tag{14}$$

where $P(k)$ is the prior probability of class k, and can be substituted by the frequency of the class among all samples.

6.3 Natural language description

We introduce a simple grammar to generate natural language descriptions from the activity patterns of tracked targets which have been recognized. Because in most surveillance scenarios, the system is often asked questions like "Who does what at where? And How?" To design a system which can answer such questions needs only a simple grammar rule. The rule is:

(The Obj) (Action) in (The place name) [***at (high/low/middle) speed***]

We integrate the map of the real scene with the activity map to fill the place name and also to establish mapping from activity to language (also called Verb selection). A typical rule like this, if target stops in the parking lot, then output action as "is parked".

6.4 Limitations

The above interpretation algorithm is not suitable to apply in a crowd scene, because the trajectories will fill the whole scene and it is very difficult to learn the typical trajectory classes. This algorithm can not handle the interactions of several targets for lack of modeling the parallel actions. We are currently working on a new framework to handle these complex situations, and will be described in our latter publications.

7 Experimental results

7.1 Demo platform

In our lab, we established a traffic demo platform which provides a test bed for our visual surveillance research. With this model scene, we can simulate accidents on this platform which are very difficult to capture in real traffic scenes.

Fig. 15 shows the traffic model scene in our demo platform which is a typical intersection. Two toy radio-controlled cars can simulate many traffic events.

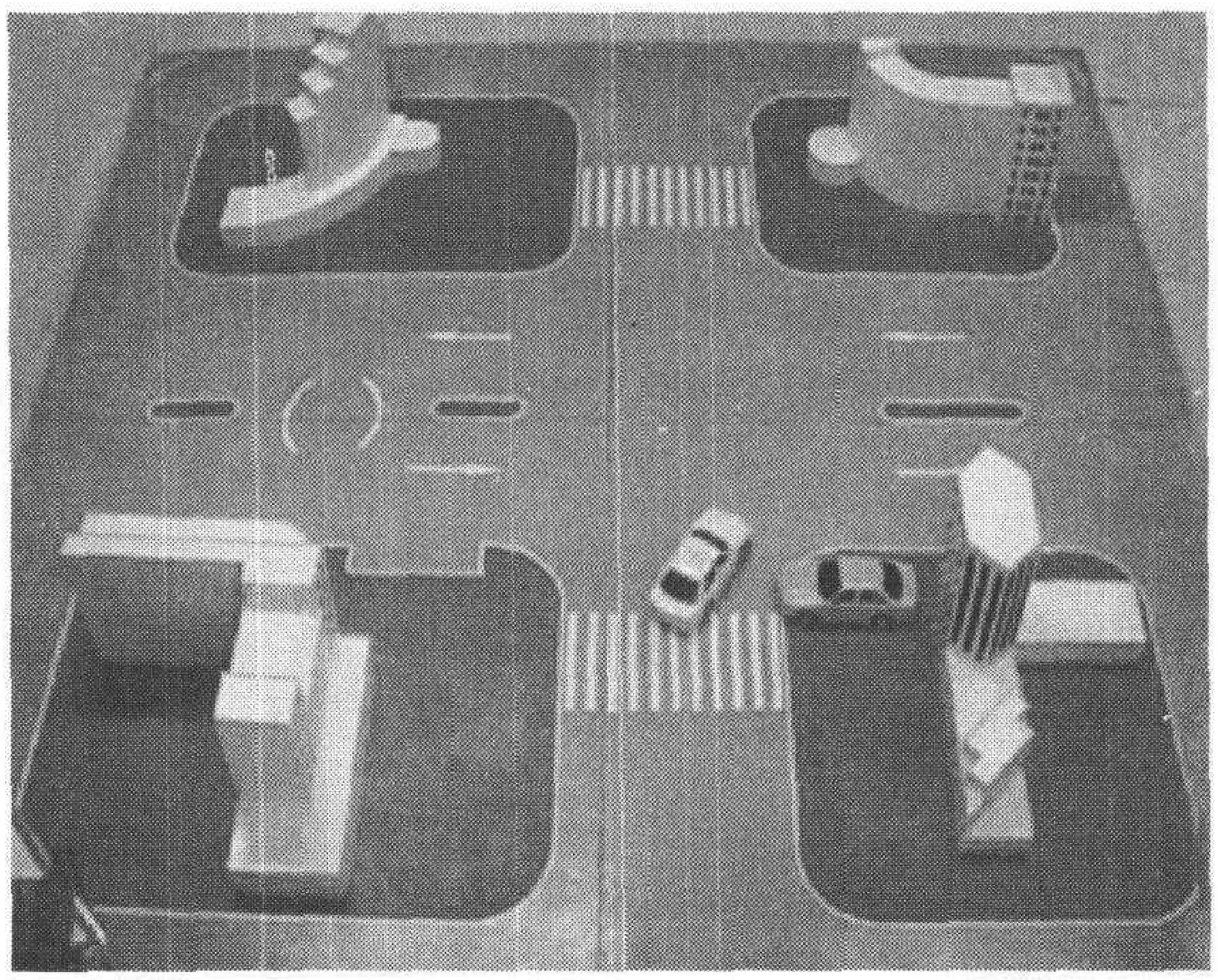


Fig. 15 The whole view of our demo scene

Some results on this demo platform are presented in Fig. 16, where the vehicle enters behind the left bottom building and turns left. The vehicle is tracked and overlapped by its 3-D wire-frame model very well at a speed of 17 frames per second. The whole sequence is presented on www. sinosurveillnace. com/yjcg2. htm.

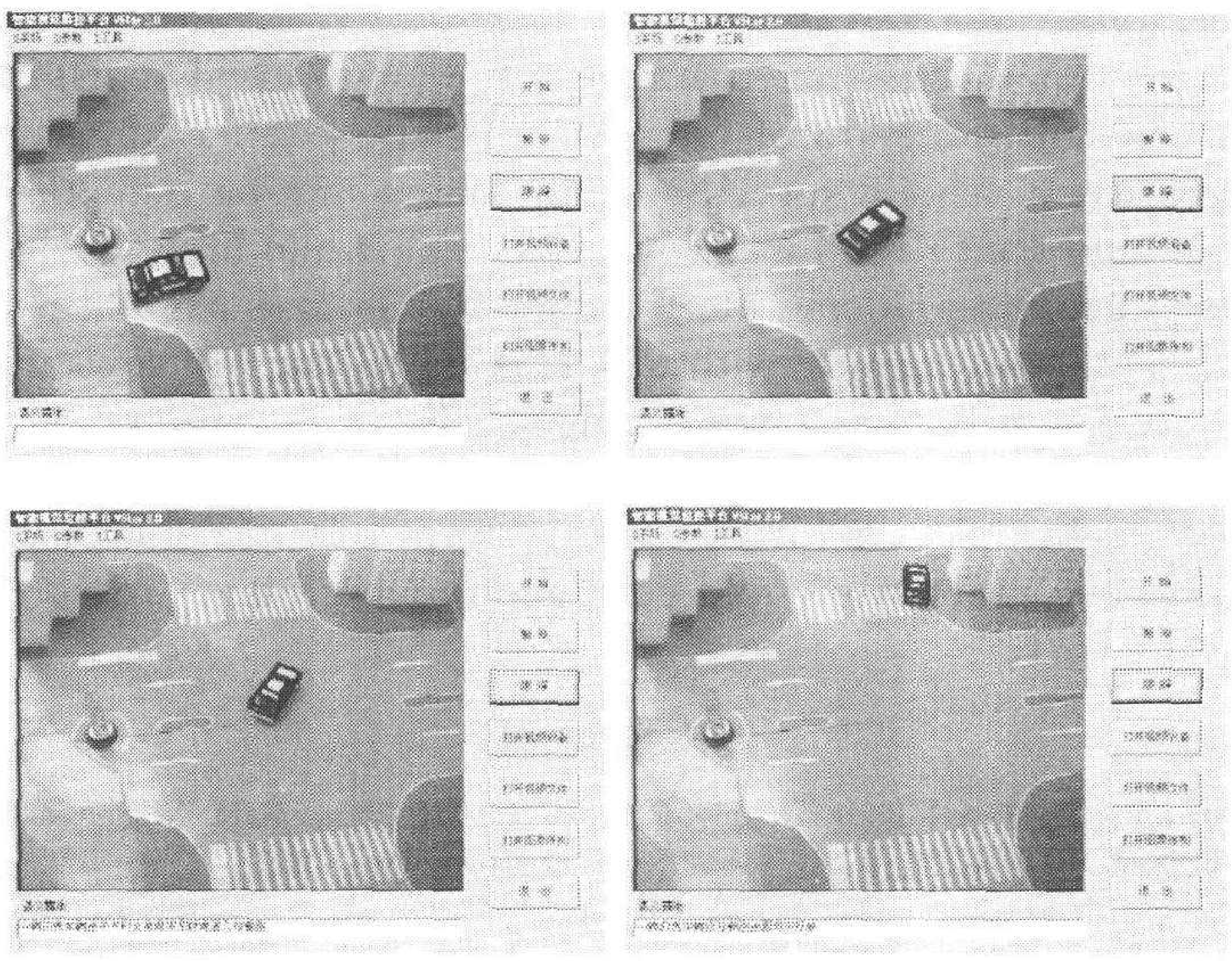


Fig. 16 A tracking result on our demo platform

7.2 Real world tracking

To demonstrate the algorithms, we also implemented them on some real world scenes. Here, we present a CASIA scene. In this scene, a black saloon is tracked. Although the car contains distinct intensity from the background, however, the skeleton of the car is not clear yet because all parts of the car are black. As we know, the perspective behind 3-D wire-frame model based algorithms is that vehicles can be presented by their skeletons intuitively. Thus, it is a big challenge for a wire-frame model based method to track a car without clear skeletons like this sequence. Fig. 17 shows a result from this scene, where the performance is quite good and the semantic interpretations are shown at the bottom of the tracking window.

Both of the sequences from the demo scene and the real world scene in Figs. 16, 17 demonstrate that all algorithms provided above can work together and obtain a good performance on efficiency, accuracy and robustness.

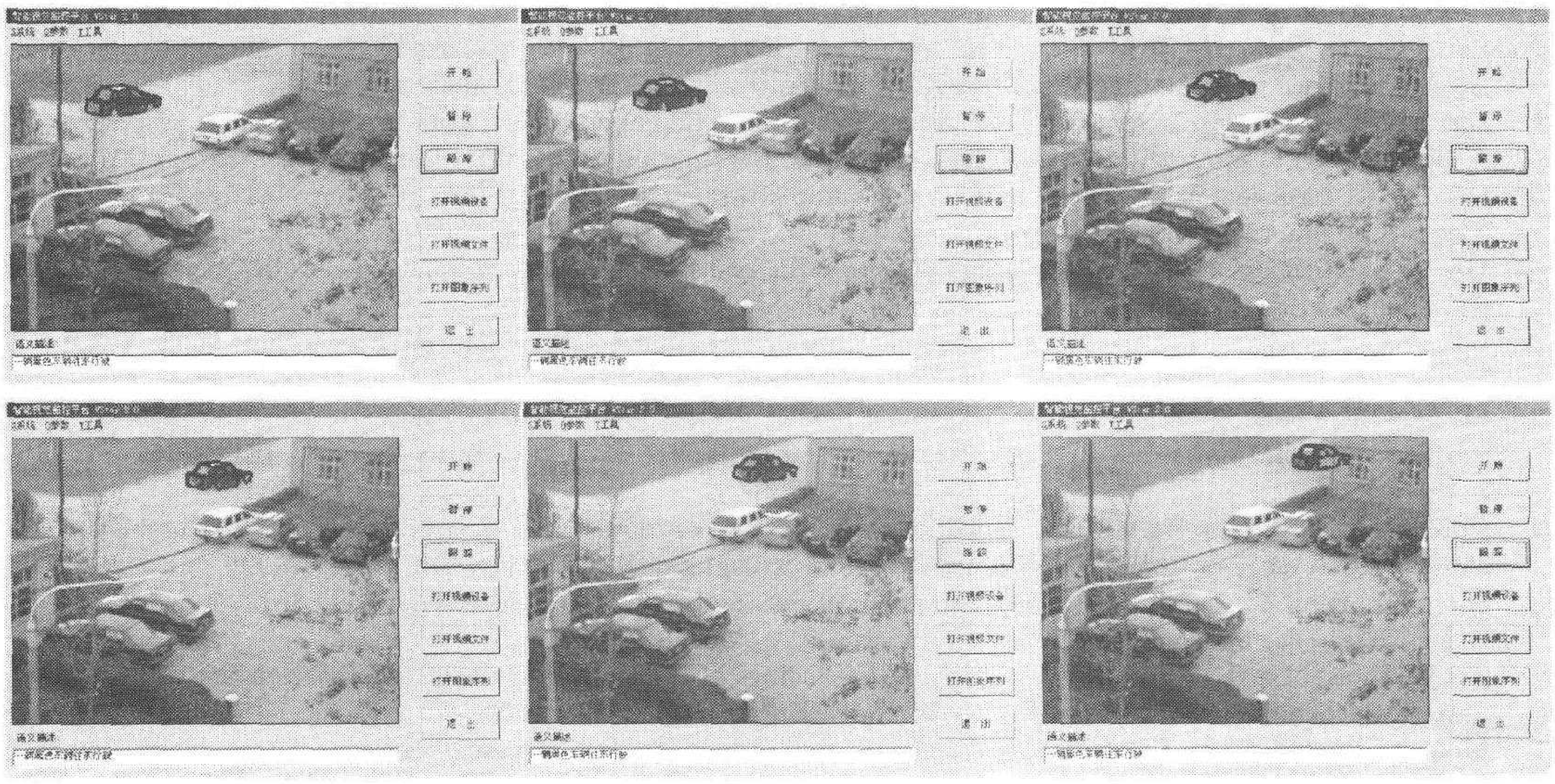


Fig. 17 A real world sequence from CASIA scene

8 Conclusions and future work

Visual traffic surveillance is an active research topic of great social importance. In this paper, we give a brief introduction of our surveillance system. In recent years, we have made many progresses on vehicle surveillance including motion detection, pose evaluation and refinement, an improved EKF for predictive tracking and high-level behaviour recognition. We also have developed a demo platform for further research which can work at a speed of 17 frames per second on a computer with PIV 1.7G CPU and Windows operating system.

However, there are still some open problems. For an example, we found that all existing pose evaluation functions fail when the target is seriously occluded or under some structural outliers[20]. In the near future, we try to integrate region information and contour information in pose evaluation, and continue our high-level research and construct a new high-level framework.

References

1 Sullivan G D, Worrall A D, Tan T N, Marslin R F, Attwood C I, Baker K D. Model based vision in VIEWS. In: ESPRIT Project Report, RU-03-FR.01, London: Reading University, 1993

2 Beymer D, Mclauchlan P, Coifman B, Malik J. A real-time computer vision system for measuring traffic parameters. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'97), Puerto Rico: IEEE Press, 1997. 495～501

3 Koller D, Daniilidis K, Nagel H H. Model-based object tracking in monocular image sequences of road traffic scenes. *International Journal of Computer Vision*, 1993, **10**(3):257～281

4 Tan T N, Sullivan G D, Baker K D. Model-based localisation and recognition of road vehicles. *International Journal of Computer Vision*, 1998, **27**(1):5～25

5 Tan T N, Baker K D. Efficient image gradient based vehicle localization. *IEEE Transactions on Image Processing*, 2000, **9**(8):1343～1356

6 Tan T N, Baker K D, Sullivan G D. 3-D structure and motion estimation from 2-D image sequences. *Image and Vision Computing*, 1993, **11**(4):203～210

7 Wells W M. Statistical approaches to feature-based object recognition. *International Journal of Computer Vision*, 1997, **21**(1):63～98

8 Lowe D G. Robust model-based motion tracking through the integration of search and estimation. *International Journal of Computer Vision*, 1992, **8**(2):113～122

9 Lou J G, Yang H, Hu W M, Tan T N. An illumination invariant change detection algorithm. In: Proceedings of the 5th Asia Conference on Computer Vision (ACCV'02), Australia: AFCV, 2002. 19～25

10 Chang Y, Hu W M, Tan T N. Model visualization in traffic surveillance. *Chinese Journal of Engineering Graphics*, 2001, Supplement: 28～33(in Chinese)

11 Yang H, Lou J G, Sun H Z, Hu W M, Tan T N. Efficient and robust vehicle localization. In: Proceedings of IEEE International Conference on Image Processing (ICIP'01), Greece: IEEE Press, 2001. 355～358

12 Liu Q F, Lou J G, Hu W M, Tan T N. Comparison of model-based pose evaluation algorithms in traffic scenes. In: Proceedings of SPIE International Conference on Image Processing(ICIG'02), Hefei: SPIE, 2002

13 Lou J G, Yang H, Hu W M, Tan T N. Visual vehicle tracking using an improved EKF. In: Proceedings of the 5th Asia Conference on Computer Vision (ACCV'02), Australia: AFCV, 2002. 296～301

14 Lou J G, Liu Q F, Hu W M, Tan T N. Semantic interpretation of object activities in a surveillance system. In: Proceedings of IAPR International Conference on Pattern Recognition (ICPR'02), Canada: IEEE Press, 2002

15 Stauffer C, Grimson W E L. Adaptive background mixture models for real-time tracking. In: International Conference of Computer Vision and Pattern Recognition, Ft. Collins: IEEE Press, 1999, **2**(2): 246～252

16 Huwer S, Niemann H. Adaptive change detection for real-time surveillance applications. In: Workshop on Visual Surveillance in ECCV2000, Ireland: IEEE Press, 2000

17 Ivanov Y, Bobick A, Liu J. Fast lighting independent background subtraction. In: Proceedings of the IEEE Workshop on Visual Surveillance, India, Bombay: IEEE Press, 1998. 49～55

18 Eveland, Konolige K, Bolles R C. Background modeling for segmentation of video-rate stereo sequences. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Santa Barbora: IEEE Press, 1998. 266～271

19 Gordon G, Darrell T, Harville M, Woodfill J. Background estimation and removal based on range and color. Internatinoal Conference of Computer Vision and Pattern Recognition, Ft. Collins: IEEE Press, 1999

20 Sun H Z, Feng T, Tan T N. Robust extraction of moving objects from video sequences. In: Proceedings of the 4th Asian Conference on Computer Vision, Taiwan: AFCV, 2000. 961～963

21 Shi J, Tomasi C. Good features to track. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'94), Seattle WA: IEEE Press, 1994. 593～600

22 Malik J, Russell S. Traffic surveillance and detection technology development: New sensor technology final report. In: Research Report UCB-ITS-PRR-97 6, California: PATH Program, 1997

23　Peter K, Karmann, Brandt A. Moving object recognition using an adaptive background memory. In: Cappellini V editor. Time-Varying Image Processing and Moving Object Recognition, Elsevier, Amsterdam, The Netherlands, 1990

24　Blake A, Isard M. Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion. Springer, 1997

25　Lowe D G. Fitting parameterized three-dimensional models to images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1991, **13**(5): 441～450

26　Brisdon, Sullivan G D, Baker K D. Feature aggregation in iconic model evaluation. In: Proceedings of Alvey Vision Conference, Manchester, 1989. 19～24

27　Borgefors G. Hierarchical chamfer matching: A parametric edge matching algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1988, **10**(6):849～865

28　Pece A E C, Worrall A D. Tracking without feature detection. In: Proceedings of PETS'2000, France: Grenoble, 2000. 29～37

29　Maybank S J, Worrall A D, Sullivan G D. Filter for car tracking based on acceleration and steering angle. In: Proceedings of British Machine Vision Conference (BMVC96), Edinburgh, UK, 1996. 615～624

30　Fernyhough J, Cohn A G, Hogg D C. Constructing qualitative event model automatically from video input. *Image and Vision Computing*, 2000, **18**(1): 81～103

LOU Jian-Guang　Received his bachelor(1997) and master degree(2000) in Department of Electrical Engineering from Zhejiang University, P. R. China. He is currently a Ph. D. candidate in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, P. R. China. His main research interests include pattern recognition, computer vision, visual surveillance, etc.

LIU Qi-Feng　Received his bachelor(1998) and master degree(2001) in Department of Computer Science from Harbin University of Science and Technology, P. R. China. He is currently a Ph. D. candidate in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, P. R. China. His main research interests include pattern recognition, computer vision, visual surveillance, etc.

TAN Tie-Niu　Received his Ph. D. degree(1989) from Imperial College of Science, Technology and Medicine, U. K. From 1989 to 1998, he joined the Computer Vision Group at the Department of Computer Science, University of Reading, where he worked as Research Fellow, Senior Research Fellow and Lecturer. Now, he is professor and director of National Laboratory of Pattern Recognition and director of Institute of Automation, Chinese Academy of Sciences. His research interests include image processing, computer vision, pattern recognition, robotics, multimedia, etc.

HU Wei-Ming　Received his Ph. D. degree(1998) in Department of Computer Science from Zheijiang University, P. R. China. from April 1998 to 2000, he worked as a postdoctoral researcher at Founder Research and Design Center, Peking University, P. R. China. Now he is an associate professor in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing P. R. China. His main research interests include neural network, computer graphics, visual surveillance, etc.

基于三维模型的交通场景视觉监控

楼建光　　柳崎峰　　谭铁牛　　胡卫明

（中国科学院自动化研究所模式识别国家重点实验室　北京　100080，中国）

（E-mail：{jglou，qfliu，tnt，wmhu}@nlpr.ia.ac.cn）

摘　要　视觉监控是计算机视觉研究的前沿方向．动态场景视觉监控就是利用计算机视觉和人工智能的理论和方法，通过对摄像机拍录的图像序列进行自动分析来对场景中的运动物体进行定位、跟踪和识别，并对物体的运动行为作出判断或者解释，达到监控的目的．本文结合交通场景监控这一特定任务，实现一个包括摄像机标定、模型可视化、运动车辆的姿态优化与定位、跟踪预测、基于轨迹分析的行为理解等功能算法的交通场景视觉监控系统．从算法和实现的角度出发，文章对系统中各个功能模块进行了较为详细的描述与讨论．

关键词　视觉监控，姿态优化，线框模型，语义理解，跟踪预测

中图分类号　TP391.41