Traffic Accident Prediction Using Vehicle Tracking and Trajectory Analysis

Weiming Hu, Xuejuan Xiao, Dan Xie, and Tieniu Tan

***Abstract*—Intelligent visual surveillance for road vehicles is a key component for developing autonomous intelligent transportation systems. In this paper, a probabilistic model for prediction of traffic accidents using 3D model based vehicle tracking is proposed. Sample data including motion trajectories are first obtained by 3D model based vehicle tracking. A fuzzy self-organizing neural network algorithm is then applied to learn activity patterns from the sample trajectories. Vehicle activities are finally predicted by locating and matching each observed partial trajectory with the learned activity patterns, and the occurrence probability of a traffic accident is determined. Experiments with a model scene show the effectiveness of the proposed algorithm.**

***Index Terms*—3D model based vehicle tracking, Activity patterns, Prediction of traffic accidents**

I. INTRODUCTION

TRAFFIC is of great importance in a modern society. The effective management of traffic, especially of road vehicles, has become an urgent problem to be solved. Traffic surveillance using monitoring cameras has already been widely applied in current traffic management. However, current methods depend on human observation of captured video sequences of images. This requires a great deal of human work and does not allow a real time response to abnormal events.

With computer vision and image processing methods, intelligent traffic surveillance systems perform localization, tracking and recognition of vehicles in video sequences captured by road cameras with little or no human intervention, and further analyze the activities of vehicles to give semantic descriptions based on the tracking results. This can facilitate daily traffic management and allow an immediate response when abnormal events occur, consequently providing a more advanced and feasible surveillance scheme.

Traffic accidents are abnormal events in traffic scenes. If a real-time system can predict accidents accurately in advance and then generate a warning, many traffic accidents may be avoided. At the same time the system can purposefully record the event as it develops. If the accident does indeed occur, the responsibility for the accident can be judged by the captured video sequences.

Some researchers have studied vision-based traffic anomaly detection that is based on motion detection [6]. Anomaly prediction is quite different from anomaly detection in that anomaly prediction aims at avoiding anomaly occurrence and anomaly detection is to detect an occurred anomaly. Anomaly prediction is more challenging.

In this paper, we study the prediction of traffic accidents using 3D model based vehicle tracking. Our work demonstrates the feasibility of a vision system to automatically predict traffic accidents. The main contributions of this paper are as follows:

- A novel framework of traffic accident prediction is presented.
- A probability model for predicting traffic accidents is constructed.

II. RELATED WORK

Visual surveillance for road vehicles generally includes three steps: motion detection, vehicle tracking, and activity understanding and description. In the following, we discuss briefly the state-of-the-art of current algorithms for 3D model based vehicle tracking, understanding and description of vehicle activities that are closely related to the work presented in this paper.

A. 3D Model based Vehicle Tracking

For visual surveillance in traffic scenes, 3D model based vehicle tracking algorithms have been studied widely. Researchers at the University of Reading [1, 2] and the University of Karlsruhe [4, 5] have contributed greatly to 3D model-based vehicle localization and tracking.

The main advantages of vehicle localization and tracking algorithms based on 3D models are:

- The introduction of prior knowledge of the 3D contour or

Manuscript received August 29, 2003. This work was supported in part by NSFC (Grant No. 60105002), Natural Science Foundation of Beijing (Grant No. 4031004) and the National 863 High-Tech R&D Program of China (Grant No. 2002AA117010 and 2002AA142100).

Weiming Hu is with National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences, Beijing, China (phone: 86-10-62551993; fax: 86-10-62551993; e-mail: wmhu@nlpr.ia.ac.cn).

Xuejuan Xiao, Dan Xie and Tieniu Tan are also with NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing, China(email: xjxiao, dxie, tnt@nlpr.ia.ac.cn).

0-7803-8125-4/03/$17.00 © 2003 IEEE

surfaces of vehicles makes the algorithms robust even under occlusion or interference between nearby image motions.

- After setting up the geometric correspondence between 2D image coordinates and 3D world coordinates by camera calibration, the algorithms naturally acquire the 3D pose of vehicles under the Ground-Plane Constraint.
- The algorithms can be applied in cases where vehicles greatly change their orientations.

Vehicle localization and tracking algorithms based on 3D models have some disadvantages such as the requirement for 3D models, high computational cost, etc.

B. Understanding and Interpretation of Vehicle Activities

Over the last decade, some efforts have been devoted to devising methods for vehicle activity understanding and interpretation [14, 15]. Fraile *et al.* [7] approximate and classify vehicle trajectories in a known ground plane. Each trajectory segment is assigned to one of the four classes: ahead, left, right, and stop. The trajectories are simplified into strings consisting of 4 symbols, and then classified by HMMs (Hidden Markov Models). Hagg and Nagel [8] employ fuzzy predicate logic calculus to interpret complex traffic scenarios. However, they believe that a probabilistic approach would provide a more suitable method to deal with the intrinsically uncertain and incomplete nature of the data provided by the images. Neumann *et al.* [9] establish a '3D scene description sequence', which includes the data detected in a traffic scene such as directions, positions and times of vehicles, etc. Then, they build up a scene framework by error-driven learning and inverse tracking in a connected network. Based on this method, Bell and Pau [10] develop an object-oriented logic program system for image interpretation and apply it to vehicle recognition in real scenes. Huang *et al.* [11] use a dynamic network structure in a visual surveillance system for highways. Remagnino *et al.* [13] present an event-based visual surveillance system for monitoring vehicles and pedestrians that supplies word descriptions for dynamic activities in 3D scenes. Jung *et al.* [18] study content-based event retrieval using semantic scene interpretation for automated traffic surveillance. Fernyhough *et al.* [23] establish the spatio-temporal region by learning the results of tracking vehicles in video sequences and construct qualitative event models by qualitative reasoning and statistical analysis.

III. OVERVIEW OF THE PROPOSED SCHEME

Our traffic accident surveillance scheme is composed of three main modules: 3D model based vehicle tracking, learning of activity patterns and prediction of traffic accidents (as shown in Fig. 1). The module for 3D model based vehicle tracking is implemented by matching the 3D vehicle models constructed in advance with the calibrated image sequences. The outputs of this module are the 3D trajectories of vehicles and the features of vehicles such as size. These outputs form the sample data for learning activity patterns. After obtaining enough sample data, we can learn the distribution of vehicle activity patterns from the data using a fuzzy self-organizing neural network. The activity patterns can be thought as the classification of vehicles' activities. In the module for traffic accident prediction, partial trajectories are matched to the learned activity patterns, and the occurrence probability of an accident is inferred from a probabilistic model. Such a probabilistic model needs to meet the following requirements:

- For two moving vehicles, we measure the matching degrees between the observed partial trajectories and all activity patterns by locating and matching the two current partial trajectories in the activity patterns.
- Using the information about current positions of vehicles, we compute the probability of vehicle collision if the two vehicles move along two trajectories corresponding to two certain activity patterns.

By analyzing the probability sequence of vehicle collision, appropriate actions can be taken to hand potential accidents (e. g. send a warning to the driver, record the scene, etc).

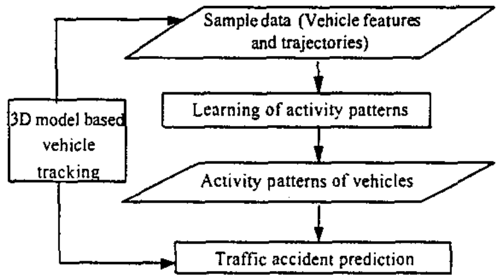

Fig. 1. Overview of the proposed traffic accident surveillance scheme

IV. 3D MODEL BASED VEHICLE TRACKING

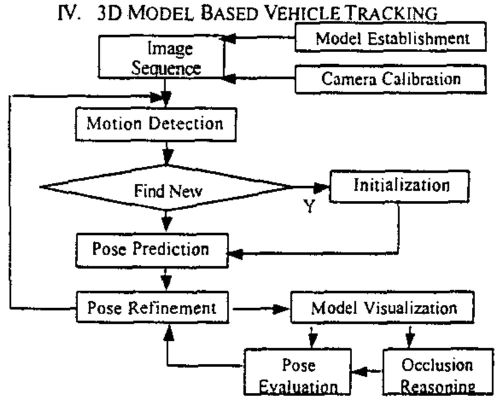

Fig. 2. Overview of 3D model based vehicle tracking.

The procedure for 3D model based vehicle tracking is shown in Fig. 2. The video captured by a calibrated camera is transformed into an image sequence. For each image, motion detection is conducted. Regions of interest (ROIs) that contain moving vehicles are detected. Each ROI is handled independently. If a new moving target is found, the tracking procedure is initialized by recognizing its vehicle type and assigning an initial value to its pose. After the initialization, the tracking takes place using pose prediction and pose refinement. Pose prediction is the estimation of the position of the vehicle in the current frame. This position is the initial value for pose refinement. Pose refinement involves a search for the optimal mapping between the projected 3D model and the 2D image data. Readers may refer to our previous paper [16, 17] for further details.

V. Learning Activity Patterns

From the information derived from the 3D model based vehicle tracking, we can acquire the training data that are used to learn vehicle activity patterns.

Our training data are composed of trajectories and features of moving vehicles. Trajectories are sampled at a fixed rate (once every Δt frames). Given a vehicle o, we represent the world coordinates of the centroid at the ith sampling as (x_i, y_i). After sampling n times, we obtain a point sequence T_o that is composed of n pairs of world coordinates: $T_o = \{(x_1, y_1), (x_2, y_2), ..., (x_i, y_i), ..., (x_{n-1}, y_{n-1}), (x_n, y_n)\}$. We use $(\delta x_i, \delta y_i)$ ($\delta x_i = x_{i+1} - x_i$, $\delta y_i = y_{i+1} - y_i$) to represent the velocity of the moving vehicle at time i. The movement of vehicle o is represented by set Q_o composed of n flow vectors: $Q_o = \{f_1, f_2, \cdots f_i, \cdots f_{n-1}, f_n\}$, where $f_i = (x_i, y_i, \delta x_i, \delta y_i)$. Similar trajectory coding schemes are used in [3, 19-22]. The features of vehicle o such as size and shape are represented by F_o, so the input data become $X_o = \{F_o, Q_o\}$. In this paper, we only consider the size of the vehicle. Naturally, the method can be easily extended to include more features such as shape, color, texture, etc. Thus the training data sampled for vehicle o are represented as $X_o = \{size, f_1, f_2, ..., f_i, ..., f_{n-1}, f_n\}$.

We use the neural network structure shown in Fig. 3 to learn vehicle activity patterns. Whole trajectory curves are used as the input to the network. In this way neurons in the input layer represent a complete trajectory whereas neurons in the output layer correspond to a class of trajectories. If there are n sampling points on a trajectory, the input vector corresponding to this trajectory includes the components:

$$(size, x_1, y_1, \delta x_1, \delta y_1, x_2, y_2, \delta x_2, \delta y_2, \cdots x_n, y_n, \delta x_n, \delta y_n).$$

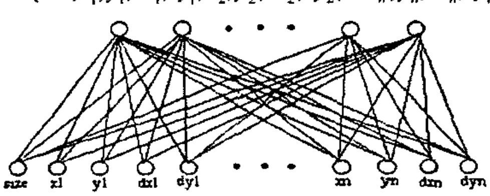

Fig. 3. Neural network structure

We use the fuzzy self-organizing neural network to train the neural network. Readers may refer to [12] for details. After the learning is completed, the activity patterns of vehicles can be represented with the output neurons. If there are K output neurons, the number of activity patterns is K. Thus we can get a set of activity patterns $\{T_i\}$, $i = 1, 2, .., K$.

VI. Prediction of Traffic Accidents

After the vehicle activity patterns are acquired, we can predict the future trajectory along which a vehicle will move using the weights of neurons according to the observed partial trajectory. According to their body shapes and sizes, we can further predict whether vehicles will come into collision or not.

Given part of a trajectory $\{(x_1, y_1), (x_2, y_2), ...(x_m, y_m)\}$, we can acquire a corresponding sub-sample:

$$X = (size, x_1, y_1, \delta x_1, \delta y_1, x_2, y_2, \delta x_2, \delta y_2, ... x_m, y_m, \delta x_m, \delta y_m)$$

The prediction of vehicle motion is obtained by computing the degree of matching between the partial sequence X and each pattern T_i in the activity pattern set $\{T_i\}$. The process to predict traffic accidents includes the following major steps:

- Locate partial test sequence X in a certain activity pattern T_i.
- Compute matching degree $P(T_i \mid X)$ between partial test sequence X and activity pattern T_i.
- Compute the probability of collision between two vehicles.

A. Locating Test Sequence X in Pattern T_i

Suppose that the length of the partial test sequence X is m and the length of pattern trajectory T_i is n ($m < n$). Since the starting point $X(x_1, y_1)$ in X may not correspond to the starting point $T_i(x_1, y_1)$ in T_i, we should find the segment in pattern T_i most similar to X. The test trajectory: $X = \{(x_1, y_1), (x_2, y_2), ..., (x_m, y_m)\}$ is actually a sequence varying with time. The last point in the test trajectory is very important to the localization task because it denotes the current position of the vehicle. Thus, we locate the test trajectory in the patterns referring to the last point $X(x_m, y_m)$. The points in the test trajectory X do not contribute equally to the matching between X and the patterns. We can see that the current point $X(x_m, y_m)$ contributes most, the $X(x_{m-1}, y_{m-1})$ second, ..., and the starting point $X(x_1, y_1)$ least. Therefore we introduce a weight $w(j)$ for each point (x_j, y_j):

$$w(j) = e^{-\frac{(j-m)^2}{m^2}}, \; j = 1, 2, ..., m. \qquad (1)$$

Let $T_{ik} = T_i(s_i, x_k, y_k, \delta x_k, \delta y_k, x_{k+1}, y_{k+1}, \delta x_{k+1}, \delta y_{k+1}, ..., x_{k+m-1}, y_{k+m-1}, \delta x_{k+m-1}, \delta y_{k+m-1})$. The weighted distance between X and T_{ik} can be defined as:

$$D_{ik} = (s_i - size)^2 + \sum_{j=1}^{m}((T_i(x_{k+j-1}) - X(x_j))^2 + (T_i(y_{k+j-1}) - X(y_j))^2 + (T_i(\delta x_{k+j-1}) - X(\delta x_j))^2 + (T_i(\delta y_{k+j-1}) - X(\delta y_j))^2) \times w(j) \qquad (2)$$

If the distance between the sub-sequence starting with point $T_i(x_k, y_k)$ and partial trajectory X becomes minimum, we choose T_{ik} as the sub-sequence in T_i which is closest to X. In other words, when partial trajectory X belongs to pattern T_i, the sub-sequence starting with point $T_i(x_k, y_k)$ best matches X.

B. Degree of Matching

Define the distance between pattern T_i and partial trajectory X as $D_i = \underset{k}{Min}\{D_{ik}\}$. The degrees of matching between T_i and X is calculated by:

$$P(T_i \mid X) = \frac{1/D_i}{\sum_{j=1}^{K}(1/D_j)}, \qquad (3)$$

where K is the total number of activity patterns and D_j is the distance between X and the sub-sequence in T_j which best matches X.

C. Prediction of collision

Suppose that the observed partial trajectory of vehicle A is X and that of vehicle B is Y. the occurrence probability in which vehicle A and vehicle B will come into collision can be calculated by the following items:

1) The probability that partial trajectory X belongs to each pattern $P(T_i \mid X)$;
2) The probability that partial trajectory Y belongs to each pattern $P(T_j \mid Y)$;
3) The probability of collision when partial trajectory X belongs to pattern T_i and partial trajectory Y belongs to pattern T_j.

Item 3 can be denoted as a function of T_i and T_j:

$$f((X, T_i), (Y, T_j)) = \begin{cases} 0 \\ 1 \end{cases}, \qquad (4)$$

where '0' denotes no collision and '1' denotes collision.

Then the occurrence probability $B(X, Y)$ of vehicle A with observed trajectory X and vehicle B with observed trajectory Y coming into collision at a later time is computed by:

$$B(X, Y) = \sum_{i=1}^{K}\sum_{j=1}^{K} P(T_i \mid X)P(T_j \mid Y)f((X, T_i), (Y, T_j)) \qquad (5)$$

where K is the size of the pattern set.

The problem here is to decide the value of the function $f((X, T_i), (Y, T_j))$. For simplicity, we represent a vehicle with a rectangular box bounding the vehicle projection on the ground plane. Thus whether the collision would happen or not can be formulated as whether the two rectangular boxes of certain sizes would intersect or not at certain time.

In Fig.4, the two solid lines represent respectively the patterns T_i and T_j in the pattern set; the solid points are sample points at equal times in the two corresponding patterns; the arrowheads show the direction of motion; the two rectangles bonded with the solid lines represent the vehicle projections at the current time; and the rectangles bounded with the dashed lines represent the vehicle projections after some time if the two vehicles move along the two patterns. Suppose that at time t, vehicle A is at position $A(x_1, y_1)$, with direction of motion $(\delta x_1, \delta y_1)$, and vehicle B is at position $B(x_2, y_2)$, with direction of motion $(\delta x_2, \delta y_2)$. The length of the rectangle for vehicle A is L_1, and the width of the rectangle is W_1. The length of the rectangle for vehicle B is L_2, and the width of the rectangle is W_2. The algorithm for judging whether vehicles A and B will come into collision is described as follows:

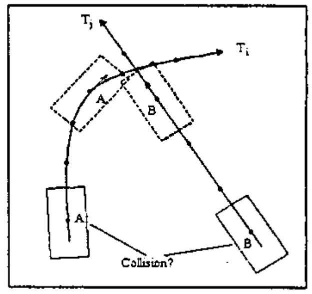

Fig. 4. Collision Judgment

t = the current time t_0;
WHILE(Sample points in patterns T_i and T_j both exist)
BEGIN
Compute intersection points between each line segment in one rectangle and each line segment in another one at time t;
IF there are n points of intersection
THEN FOR each point of intersection
BEGIN
Compute distance d_1 between the point and $A(x_1, y_1)$;
Compute distance d_2 between the point and $B(x_2, y_2)$;
IF $(d_1 < \sqrt{L_1^2 + W_1^2}/2)$ or $(d_2 < \sqrt{L_2^2 + W_2^2}/2)$
THEN RETURN collision;
END
$t = t$ + the sampling time;
END
RETURN no collision;

In fact, it is not accurate enough to judge vehicle collision with Formula (4) when partial trajectories X and Y belong respectively to patterns T_i and T_j. The probability of collision is related to time t when the collision may happen, and current time t_0. The longer the predicted collision time t is away from current time t_0, the less likely the collision. On the contrary, the closer time t is to current time t_0, the more likely the collision. So when the above geometric condition of vehicle collision is satisfied, we introduce an weight function to Formula (4) which is thus improved in the following fashion:

$$f((X, T_i), (Y, T_j)) = \begin{cases} 0 \\ e^{-\frac{(t-t_0)^2}{2\sigma^2}} \end{cases} \qquad (6)$$

where '0' stands for no collision, and $e^{-\frac{(t-t_0)^2}{2\sigma^2}}$ is the probability of collision. Assume that the driver's response time is h fames, that is to say, if the driver knows the danger h frames ahead, the collisions might be avoided. We consider that the probability distribution should be changed fast at this time, so $\sigma = h$.

VII. EXPERIMENTS

All the algorithms were implemented using Visual C++ 6.0 on the Windows 2000 platform Since it is difficult and dangerous to capture or simulate accidents in real scenes, we have carried out our experiments in an indoor model of a real traffic scene (as shown in Fig.5). The model is of size 2.4m*2.4m. The model includes a crossroads, parking lots, one-way roads and multilane roads, etc. Furthermore it also involves many events such as turning left, turning right, entering and leaving. The model also includes radio-controlled toy cars. Because the algorithms for vehicle localization and tracking are based on 3D models, we have also made 3D wire-frame models for the toy cars.

Fig. 5. A model traffic scene

We have implemented a real-time 3D model based vehicle tracking system in the traffic model scene. By driving the toy cars, we can acquire a series of trajectories. By learning these trajectories, we can obtain the activity patterns that are used to realize the prediction of car collision. Since all experiments are based on 3D model based tracking, all the data used in the experiments are in a single world coordinate system In order to show the experiment results more intuitively, the trajectories in the following figures overlaid in the images.

Fig. 6 shows 400 trajectories acquired by 3D model based vehicle tracking. The learning results of activity patterns are shown in Fig. 7, in which the black lines represent the trajectory samples and the white ones denote the learned activity patterns. The results appear to be satisfactory since there are no oscillations and the learned patterns are consistent with the samples.

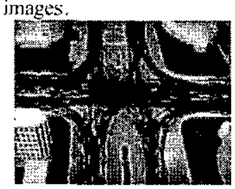

Fig. 6. Trajectory samples

Two test instances are shown in Fig. 8 and Fig. 9 in which the vehicles are tracked accurately using 3D wire-frame vehicle models. The image sequences are sampled once every three frames. The figure for frame 45 in Fig. 8 is at a larger size in order to show the projected wire-frame model more clearly.

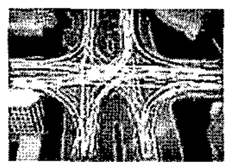

Fig. 7. Learned activity patterns

Table 1 shows the occurrence probability of collision for test image sequence 1. The first row shows the results of Formula (5) with Formula (4), and the second row shows the results with Formula (6) to which the weight function is introduced. Test image sequence 1 shows a case in which two vehicles come close but do not collide. In Table 1, the probability shown in row 1 begins to decrease from frame 33, whereas that shown in row 2 begins to decrease obviously from frame 30. This shows that the probability with weight analysis is better for prediction than that without weight analysis since the two vehicles do not at last collide. But it is still very dangerous that the vehicles in the test sequence approached each other so closely. Thus at frame 27, the probabilities in two rows are both above 70%, which can serve as a warning indicator.

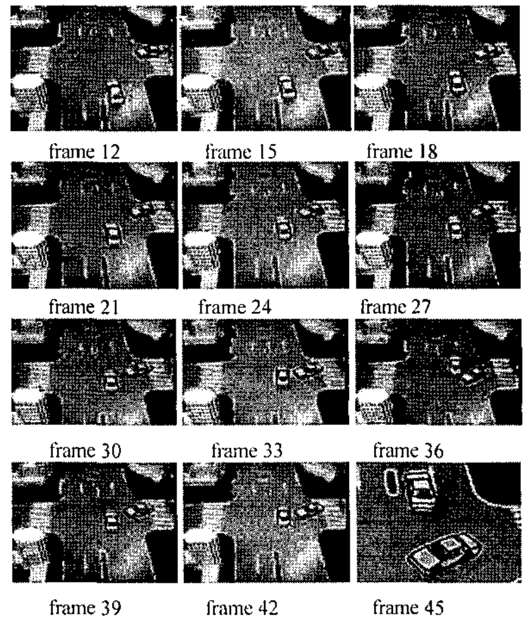

Fig. 8. Image sequence 1 for test

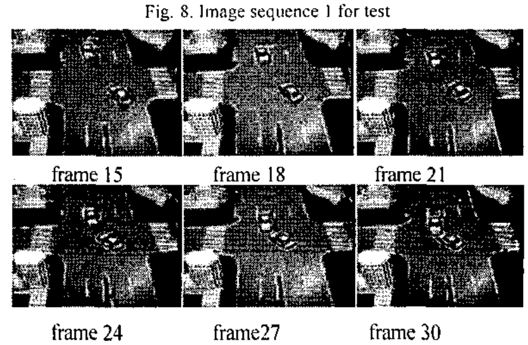

Fig. 9. Image sequence 2 for test

Table 2 shows the probabilities of vehicle collision in Fig. 9, where collision actually occurred. The two rows show the probabilities with and without weight analysis respectively. We can see that, just before the collision occurs, the probabilities with weight analysis observed in frames 27 and 30 are larger than those without weight analysis. So the possibilities with weight analysis are also better for prediction than those without weight analysis.

Fig. 10 shows the experimental results for other 7 image sequences. The solid curves represent the probability sequences corresponding to the cases in which collisions happen. The dash-dot curves correspond to the cases in which collisions do not happen but the vehicles come close to each other. The dotted curves correspond to the cases in which no danger exists. Because in the initial frames there is too little information, the collision probabilities predicted in the initial sampling phase are instable. In practice, we just ignore the first few frames. We set a threshold to discriminate between the normal and dangerous situations, shown as line b in Fig. 10. When the probability of collision is above a predefined threshold, collision is predicted.

Table 1. The occurrence probability of collision for Fig. 8 (%)

Frame no. / Formula	12	15	18	21	24	27	30	33	36	39	42	45
Formula (4)	31	38	34	44	60	75	72	44	31	0	0	0
Formula (6)	31	34	34	43	57	74	62	52	21	0	0	0

Table 2. The occurrence probability of collision for Fig. 9 (%)

Frame no. / Formula	12	15	18	21	24	27	30
Formula (4)	37	42	38	48	54	62	82
Formula (6)	38	42	37	46	50	72	84

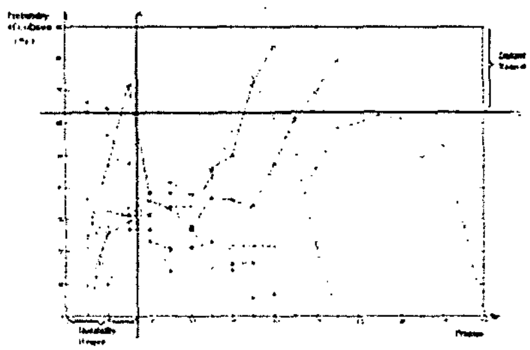

Fig. 10. Analysis of traffic accidence prediction

VIII. CONCLUSIONS

We have presented a probability model for traffic accident prediction. We first obtain sample data including motion trajectories by 3D model based vehicle tracking. We then establish a probability model for predicting traffic accidents. We predict the trajectory along which a vehicle will move by matching its current partial trajectory with the learned activity patterns, and further calculate the possibility in which two vehicles will collide. Experiments for traffic accident prediction are performed based on the 3D model based vehicle tracking system. The results demonstrate the effectiveness of the proposed algorithms.

REFERENCES

[1] T. N. Tan, G. D. Sullivan and K. D. Baker, " Model-based localization and recognition of road vehicles", International Journal of Computer Vision, Vol. 27, No. 1, pp. 5-25, 1998.

[2] T. N. Tan and K. D. Baker, " Efficient image gradient based vehicle localization, IEEE Trans. on Image Processing, Vol. 9, No. 8, pp. 1343-1356, 2000.

[3] N. Johnson, " Behaviour Model and Analysis", Ph. D. Thesis, The University of Leeds, 1999.

[4] H. Kollnig and H-H. Nagel, " 3D pose estimation by directly matching polyhedral models to gray value gradients" , International Journal of Computer Vision, Vol. 23, No. 3, pp. 283-302, 1997.

[5] M. Haag and H-H. Nagel, "Combination of edge element and optical flow estimates for 3D-model-based vehicle tracking in traffic image sequences", International Journal of Computer Vision, Vol. 35, No. 3, pp. 295-319, 1999.

[6] M. Trivedi, I. Mikic, G. Kogut, " Distributed video networks for incident detection and management" , IEEE Conference on Intelligent Transportation Systems, Dearborn, Michigan, October 2000

[7] R. Fraile and S. J. Maybank, " Vehicle Trajectory Approximation and Classification" , Proceedings of British Machine Vision Conference, pp. 832-840, 1998.

[8] M. Hagg and H. -H. Nagel, "Incremental recognition of traffic sequences", ICCCV-98 Workshop on Conception Description of Images, pp. 1-20, 1998

[9] M. Mohnhaupy and B. Neumann, " On the use of motion concepts for top-down control in traffic scene" , Proceeding of European Conference on Computer Vision, pp. 542-550, 1990.

[10] B. Bell and L. F. Pau, " Context knowledge and search control issues in object-oriented Prolog-based image understanding" , Pattern Recognition Letters, Vol. 13, No. 4, pp. 279-290, 1992.

[11] T. Huang, D. Koller, J. Malik, G. Ogasawara, B. Rao, S. Russell, and J. Weber, " Automatic symbolic traffic scene analysis using belief networks" , Proceedings of 12th National Conference on Artificial Intelligence, Seattle, WA, pp. 966-972, 1994.

[12] Weiming Hu, Dan Xie and Tieniu Tan, " Learning patterns of activity using fuzzy self-organizing neural network" , IEEE Transactions on System, Man and Cybernetics, Part B, Under Second Review

[13] P. Remagnino, T. N. Tan, A. D. Worrall and K. D. Baker, " Multi-agent visual surveillance of dynamic scenes", Image and Vision Computing, Vol. 16, No. 8, pp. 529-532, 1998.

[14] D. Koller, N. Heinze and H. -H. Nagel, "Algorithmic characterization of vehicle trajectories from image sequences of motion verbs", Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 90-95, 1991.

[15] M. Mohnhaupt, B. Neumann, "On the use of motion concepts for top-down control in traffic scenes", Proceedings of European Conference on Computer Vision, pp. 598-600, 1990

[16] Hao Yang, Jianguang Lou, Hongzan Sun, Weiming Hu, Tieniu Tan, " Efficient and robust vehicle localization" , IEEE International Conference on Image Processing, pp. 355-358, 2001.

[17] Jianguang Lou, Hao Yang, Weiming Hu, Tieniu Tan, " Visual vehicle tracking using an improved EKF" , Asian Conference on Computer Vision, pp. 296-301, 2002.

[18] Y. K. Jung, K. W. Lee and Y. S. Ho, " Content-based event retrieval using semantic scene interpretation for automated traffic surveillance" , IEEE Trans. on Intelligent Transportation Systems, Vol. 2, No. 3, pp. 151-163, 2001

[19] N. Johnson, and D. Hogg, " Learning the Distribution of Object Trajectories for Event Recognition" , Image and Vision Computing, Vol. 14, No. 8, pp. 609-615, 1996.

[20] J. Owens and A. Hunter, " Application of the self-organizing map to trajectory classification", Proceedings of IEEE Workshop on Visual Surveillance, pp.77-83, 2000.

[21] C. Stauffer, W. Eric L. Grimson, " Learning patterns of activity using real-time tracking" , IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 22, No. 8, pp. 747-757, 2000.

[22] N. Sumpter and A. Bulpitt, " Learning spatio-temporal patterns for predicting object behavior", Image and Vision Computing, Vol. 18, No. 9, pp. 697-704, 2000.

[23] J. Fernyhough, A. G. Cohn and D. C. Hogg, " Constructing qualitative event models automatically from video input" , Image and Vision Computing Vol. 18, No. 9, pp. 81-103, 2000.