Lane Detection and Car Tracking on the Highway[1)]

ZENG Zhi-Hong
(*National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing* 100080, *P. R. China*)
(E-mail: zhzeng@liama. ia. ac. cn)

Abstract An efficient vision system is proposed for lane detection and multiple car tracking on the highway. The main modules of the system are: lane detection, separate 2-D model-based trackers (rectangular model for passing car and U-shape model for distant car), heuristic car detection, and a process coordinator. In the system, the dynamical creation and termination of tracking processes optimizes the amount of spent computational resources. Lane detection performance is improved by robust estimation technique. And car tracking is realized by a polygon fitting approach in three-parameter state space. The system is successfully tested with the image sequence from PETS2001 and the average processing time per frame is 12ms on a Pentium Ⅲ 450MHz PC.

Key words Lane deteition, car tracking, edge points, polygon fitting

1 Introduction

Road following is a complex and challenging task and has received most attention in Automatic Vehicle Guidance[1~8]. Its two major components are lane detection and obstacle detection[1]. Lane detection, generally, can be reduced to the localization of specific features such as lane markers painted on the road surface. And obstacles on the road are mainly vehicles.

This paper presents an efficient vision system for lane detection and multiple car tracking on the highway. The main modules of the system are: lane detection, separate 2-D model-based trackers, heuristic car detection, and a process coordinator. In the system, the dynamical creation and termination of tracking processes optimizes the amount of spent computational resources.

In the system, we take advantage of the spatial-temporal continuity of lane markers in sequences to improve the efficiency and robustness of lane detection. Lane detection performance is improved by robust straight line estimation technique. Based on the good results of lane detection, car detection can achieve the promising performance.

Many efforts have been made to apply model-based methods to track vehicles in recent years[6,9,10]. Model-based algorithms have a level of stability and accuracy that is impossible to achieve with image-based trackers. However, when the car is moving on the highway and the input image sequence is like in Fig. 3, both the simple 2-D rectangular box model[5] and the weak perspective assumption in 3-D model methods[9,10] are invalid so that tracking becomes very complicated.

In order to deal with the above problem in an efficient way, a separate 2-D model-based tracking approach is proposed in this paper. Based on the position of the tracked car relative to the camera, a different 2-D car model is used to track the car. Then, the search process is reduced to a polygon fitting process in the three-parameter space. The method appears more robust than the low-level feature extraction method such as [5], and more efficient than 3-D model-based methods such as [9,10].

The image sequence provided by PETS2001 is used to test the system. The perform-

1) Supported by the Digital Museum Project and the National Natural Science Foundation of P. R. China(60135020)
Received October 5, 2002; in revised form March 25, 2003
收稿日期 2002-10-05; 收修改稿日期 2003-03-25

ance of the system is promising and the average processing time per frame is 12ms on Pentium Ⅲ 450MHz PC.

2 System framework

Our system contains four main modules: lane detection, car detection, multiple trackers and a process coordinator. The system is outlined in Fig. 1. The lane detection module is responsible for detecting lane markers painted on the road and then separating the road area from the scene. The car trackers are responsible for refining tracking results at preceding frames by a fitting technique. The car detection module is responsible for detecting cars that enter the road in images. The process coordinator is responsible for starting and terminating a tracker, setting the detection regions based on the lane detection, and switching between the two separate models during tracking. If car detection module detects an entering car in images, the process coordinator creates a new tracker for the potential car. Otherwise, if a tracker meets the termination conditions, process coordinator terminates the tracker.

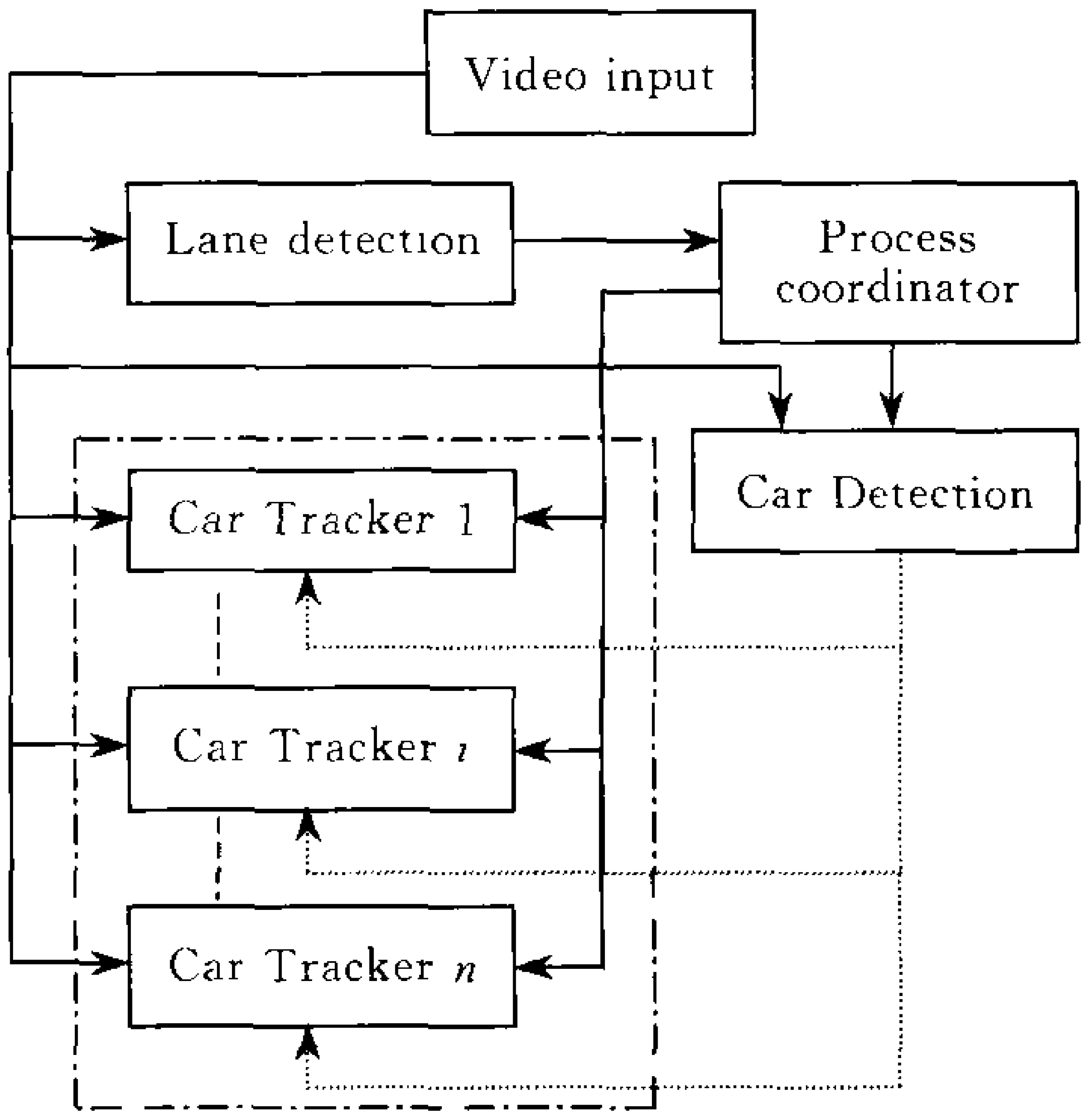


Fig. 1 The vision system framework

3 Lane detection

The roadway is modeled as a flat surface with white markers painted on it. The first stage of the lane detection is to detect and localize possible lane markers on each row of the input image. The lane markers are always referred to white bars of a particular width against a darker background. In our system, only the two lane lines just in front of the camera are detected since the markers on these lines are generally visible. The whole road area can be derived from the result of the lane detection. Based on the fact that the curvature of a highway is generally small, straight lines can be used to approximate a certain range of lanes on the highway. Once a set of lane marker points have been extracted, the robust straight line estimation[11] is used to estimate the lane in images. The robust estimation[11] makes the algorithm insensitive to a few outlier points. In order to further resist the disturbance from marker occlusion by other vehicles, shadows or cracks on the roadway, the following measures are taken in our system:

1) On each row of the input image, the search regions for the markers are based on the detected lanes at the preceding frame and their search ranges increase linearly with the road widths on the image rows, shown as white regions in Fig. 2.

2) The marker edge points are defined as the points in the search regions with the strongest signed vertical edge strengths larger than a preset threshold, shown in Fig. 2. The extracted marker edge points are used to compute the fitted straight lines.

3) In the extreme case, if the number of extracted marker edge points is too small due to marker occlusion, the fitting process halts and the results at the preceding frame are used. Generally, only one of two lane lines is severely occluded. At this time, the width between the two lane lines at the preceding frame can be used to deduce the occluded lane line.

The first and third measures are based on the spatial-temporal continuity in image sequences. The results of lane detection are shown as white oblique lines in the Fig. 2.

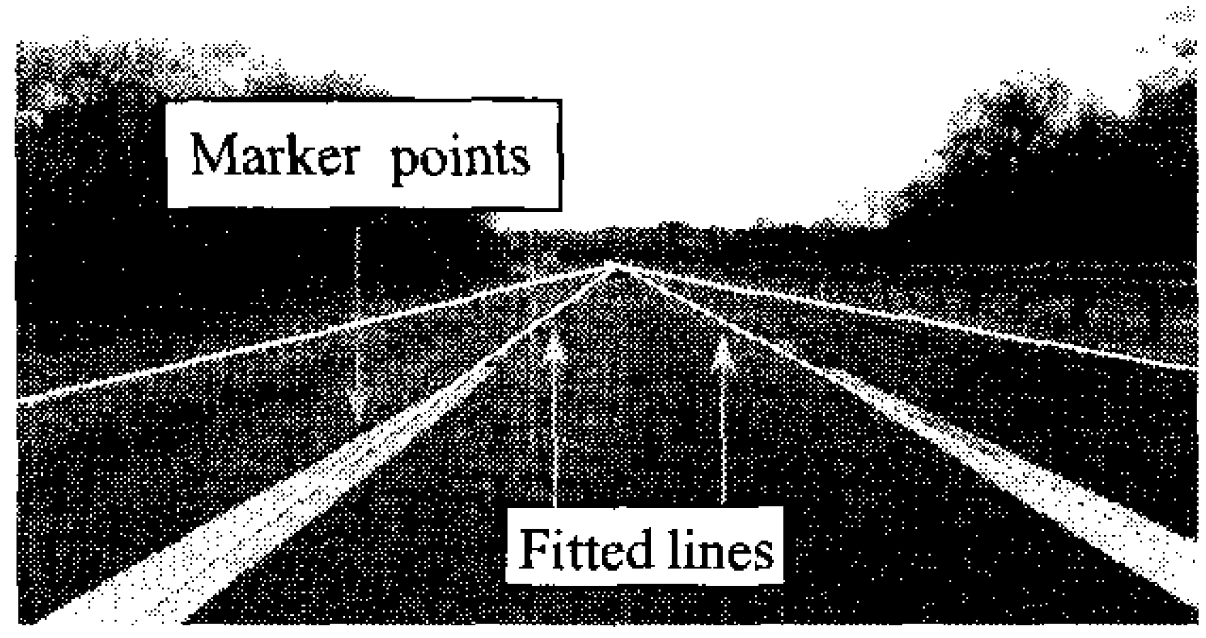


Fig. 2 Lane detection. The white regions are for detecting lane markers. The two middle oblique white lines are the line fitting results. The other oblique white lines are deduced by the fitting results and prior knowledge of road structure

4 Separate models and switch line

We observe that when the car is in the distance, the rear of the car is mainly visible which is like rectangular shape; when the car is sideward in the image and close to the camera, the side of the car is mainly visible. Based on such an observation, we propose the following separate 2-D models for the two different cases:

1) Model 1. rectangle model with four borders is for the side of the passing car which is close to the camera.

2) Model 2. U-shape model with three borders is for the rear of a car in the distance or just in front of the camera. We adopt the U-shape model rather than the rectangle is based on the fact that cars have various heights, and the top borders of cars cannot be reliably detected due to the disturbance in the background such as the horizon in the distance like in Fig. 3. These two models are shown in Fig. 3.

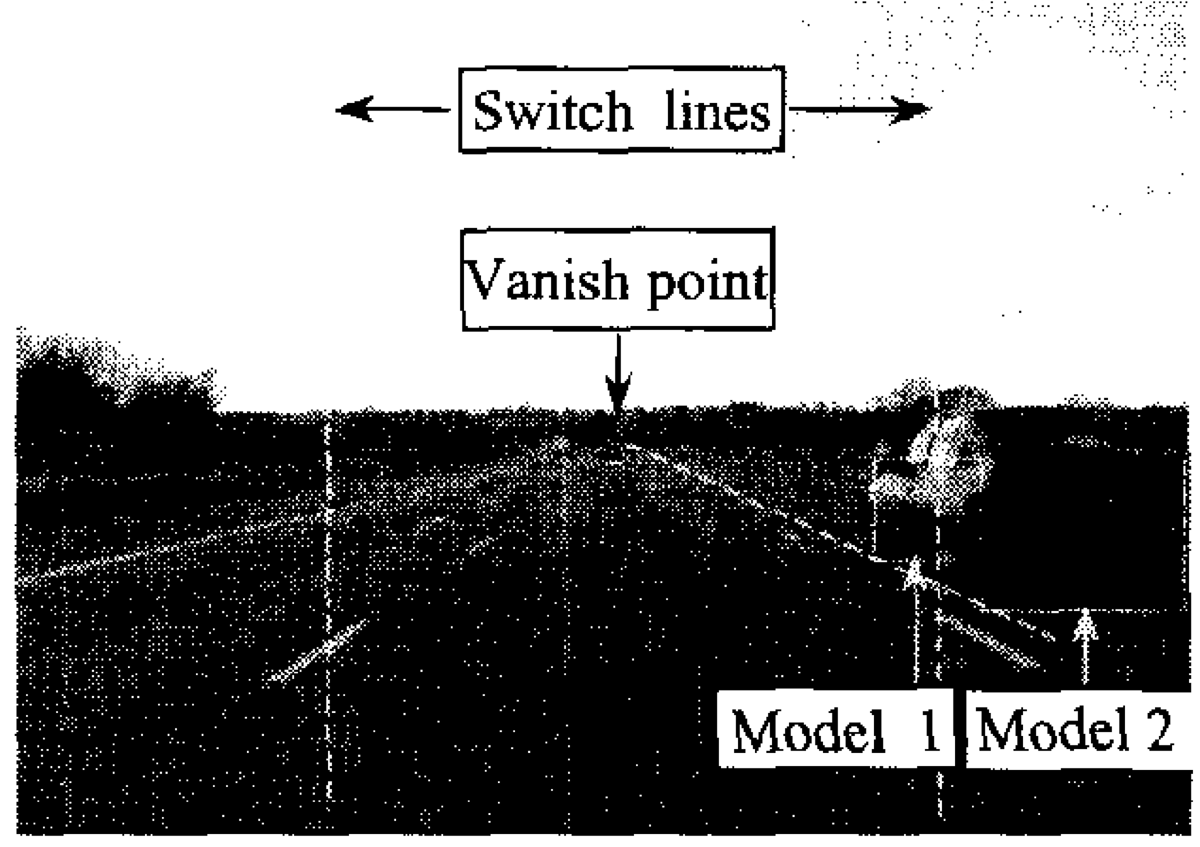


Fig. 3 The white rectangle and U shape represent the Model 1 and Model 2 respectively. The vertical grey dashed lines are the switch line from Model 1 to Model 2. The vanish point is the intersection of lane lines. The oblique grey dashed line is defined by the vanish point and the left-bottom corner of Model 1

The coordinate system of the image is set as: x axis is the horizontal axis, y axis the vertical axis, and the origin is at the left bottom of images. The x mean value of the corners of the tracked car in the image determines which one of the two models is used in tracking process. There are two switch lines (vertical dased lines in Fig. 3) through which the tracker with Model 1 switchs to the one with Model 2. In Fig. 3, the tracker goes through the right switch line, so Model 1 switchs to Model 2 whose initial parameters are: 1) the left-bottom corner of Model 2 is the intersection of the right border line of Model 1 and the straight line defined by the vanish point and the left-bottom corner of Model 1 (the oblique dased line in Fig. 3); 2) the width and height of Model 2 are initialized as 100 and 120 pixels respectively. Finally, the exact parameters of Model 2 are calculated by the following fitting technique.

5 Car tracking

When a proper model is selected, the tracking process is conducted by a fitting technique. In this paper, the two kinds of models are considered as polygons. The U-shape model is viewed as an open polygon. Therefore, we apply the polygon fitting technique to update the tracking results frame by frame.

We first define a space S_Q of control vectors Q consisting of polygonal corner coordinates.

$$Q = \begin{pmatrix} Q^x \\ Q^y \end{pmatrix} \tag{1}$$

where $Q^x = (q_0^x q_1^x \cdots q_{N_B-1}^x)$ and $Q^y = (q_0^y q_1^y \cdots q_{N_B-1}^y)$. N_B is the number of the polygonal corners and $q_n = (q_n^x, q_n^y)$ is the corner coordinates.

Then, we define the shape space $S = L(W, Q_0)$, which is a linear mapping of a shape space vector $X \in R^{N_x}$ to a polygon vector $Q \in R^{N_q}$:

$$Q = WX + Q_0 \tag{2}$$

where Q_0 is the space vector of the template polygon, and W is a shape-matrix

$$W = \begin{pmatrix} 1 & 0 & Q_0^x \\ 0 & 1 & Q_0^y \end{pmatrix} \tag{3}$$

The first two columns of W govern horizontal and vertical translations respectively. The third column controls scaling. Therefore, the transformation of the polygon is described by three-parameter vector $\boldsymbol{X}$.

If the fitted polygon $\boldsymbol{r}$ is restricted to shape space and biased towards a predicted polygon $\bar{\boldsymbol{r}}$ to a degree determined by a regularization constant α, the fitted curve $\boldsymbol{r}$ is the solution of

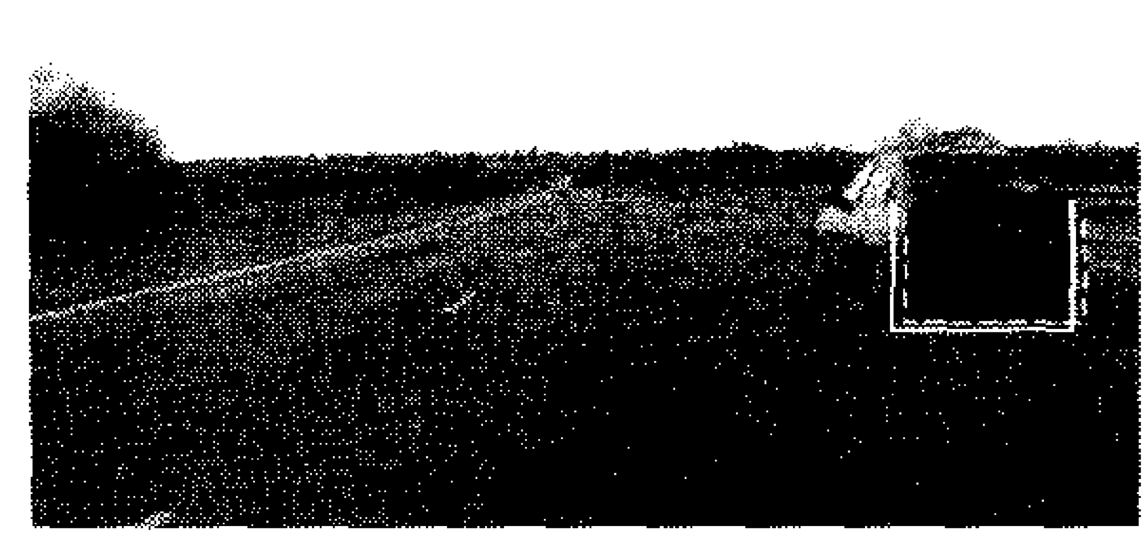
$$\min_{r(x)} \alpha \| \boldsymbol{r} - \bar{\boldsymbol{r}} \|^2 + \| \boldsymbol{r} - \boldsymbol{r}_f \|^2 \tag{4}$$

where $\boldsymbol{r}_f$ is an image-data polygon.

The fitting process is based on the normal displacement along the predicted contour. That is illustrated in Fig. 4. The white dashed and solid polygons are the predicted curve and the fitted one respectively. The algorithm recursively updates the estimated shape X by traversing the data along the predicted contour.

A car tracker is terminated if the tracker meets the following conditions: the height of the shape of the tracked car is smaller than 10 pixels, or the x mean value of corners of the tracked car is smaller than 30 pixels or larger than the width of the image width minus 30 pixels. The above terminating conditions mean that the tracked car will disappear in the distance or sideward.

(a)



(b)

Fig. 4 Tracking process. The white dashed and solid rectangles (a) or U shapes (b) represent the predicted and fitted shapes respectively. (a) and (b) are Model 1 and Model 2 tracking results respectively

6 Car detection

Based on lane detection, the detection regions are set as a triangular region and two side detection bars shown in Fig. 5. At every frame, after car tracking is finished, the areas occupied by tracked cars can be cut off from the detection areas. Then, car detection is processed on the free area in the detection regions.

In the triangular detection region, vertical edge points are extracted and accumulated along the vertical direction. If the largest number of the accumulated edge points is larger than a threshold, fitting process is started with Model 2. Such an extracted vertical line could belong to either left side or right side of a U-shape model. Therefore, both the left U-shape model and right U-shape model are used to initialize the fitting process. Then the two fitting results are compared and the one with stronger horizontal edge strength along the bottom side of the U-shape model is selected as the detection result. Later, a new tracker is started.

In a side detection bar, if the number of edge points is larger than a preset threshold, a new tracker is started with Model 1.

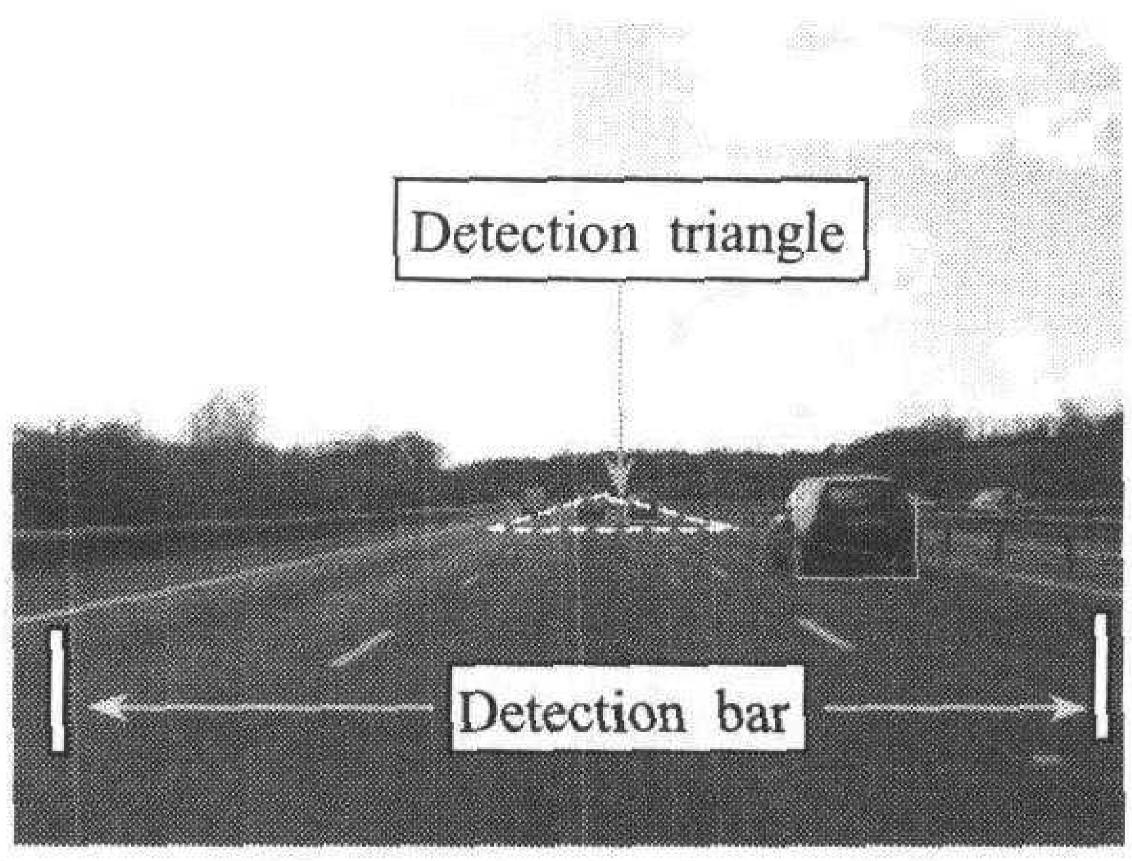


Fig. 5 Detection regions in Frame 1933 include a triangular region and two bars

7 Analysis of experimental results

Our test image sequence data is from the PETS2001 datasets: //www. visualsurveillance. org/PETS2001.

The sequence consists of 2786 images whose resolutions are 768×576 pixels. In the whole sequence, cars enter and disappear in the distance or sideward. At the end phase of the sequence, the camera-equiped vehicle turns right from the middle lane to the right lane. The system successfully realizes lane detection, car detection and tracking across the whole sequence. Some outputs of the system are shown in Figs. 6~9. Worth of mention is that we encounter zero false alarm in the sequence. False alarm means that the system detects a car on the location where no car exists. That is due to the good results of lane detection. The average processing time per frame is 12ms on a Pentium Ⅲ 450Mhz PC.

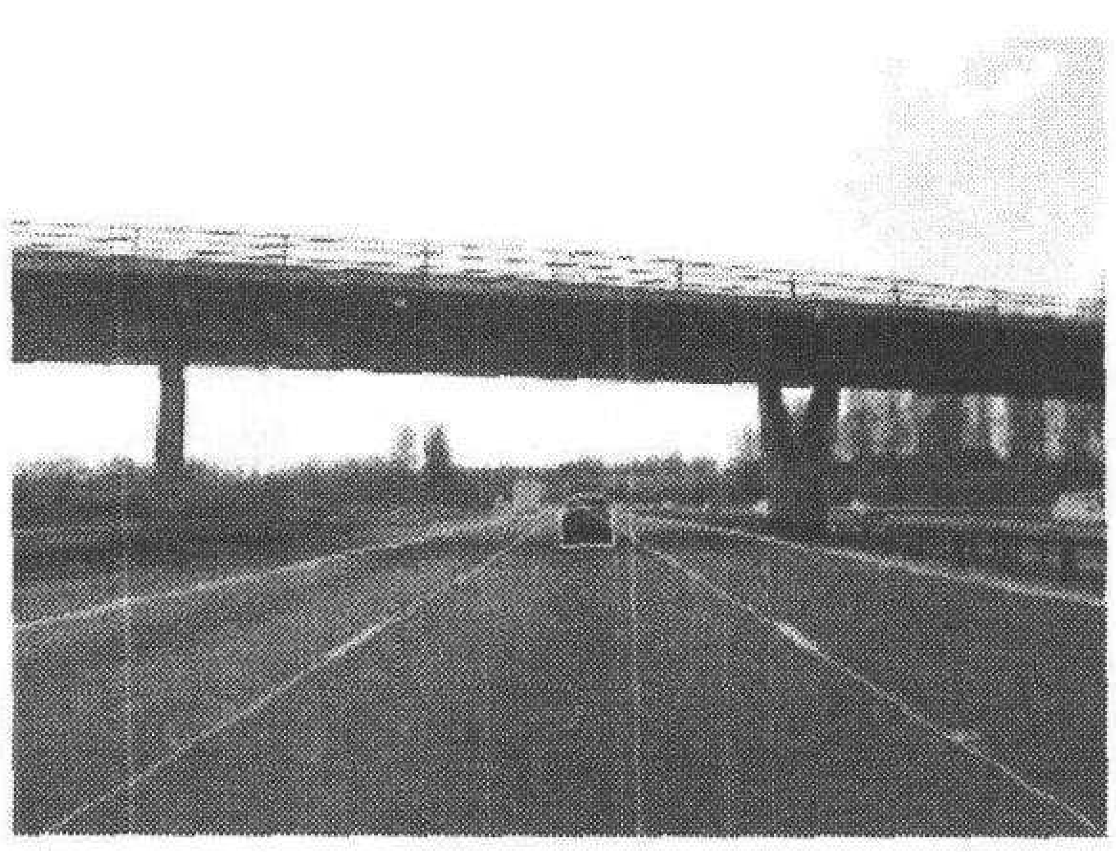


Fig. 6 Outputs of the system at frame 1474. The car is running under a bridge. The white U shapes are car tracking results. The white straight lines are lane detection results

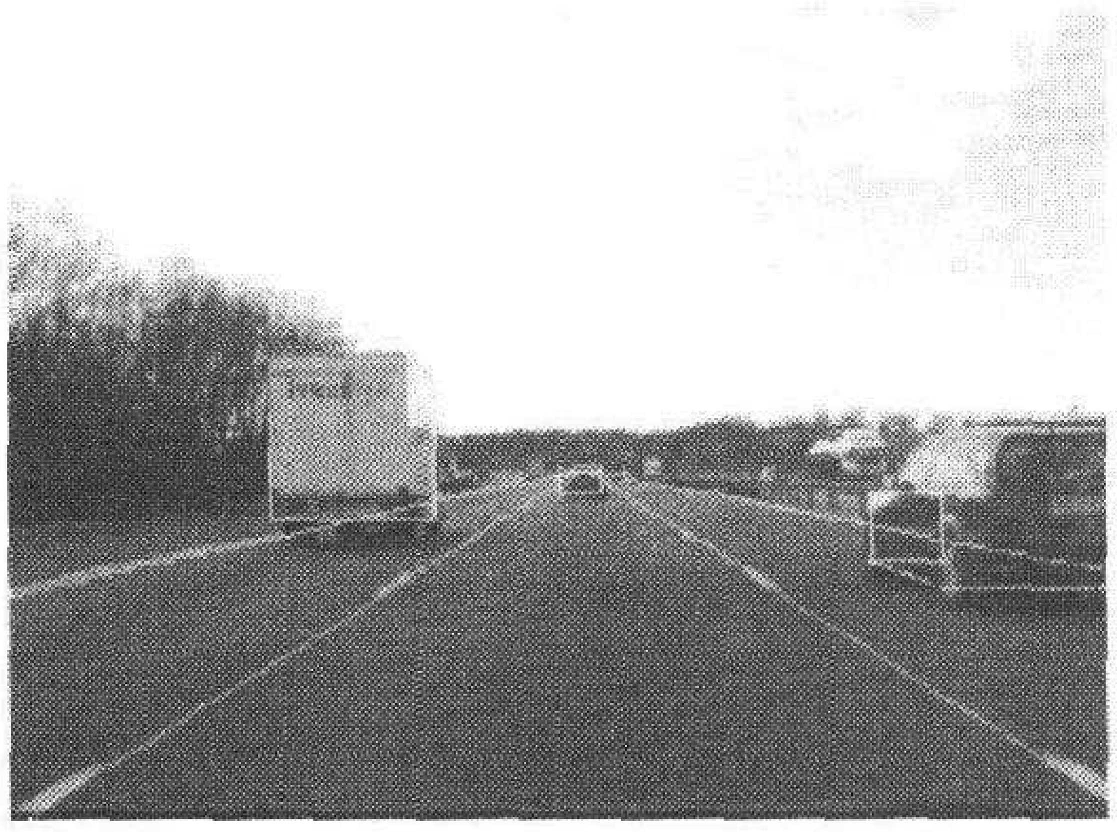


Fig. 7 Outputs of the system at frame 1734. Multiple cars are tracked simultaneously. The white U and rectangle shapes are the tracking results. The white straight lines are lane detection results

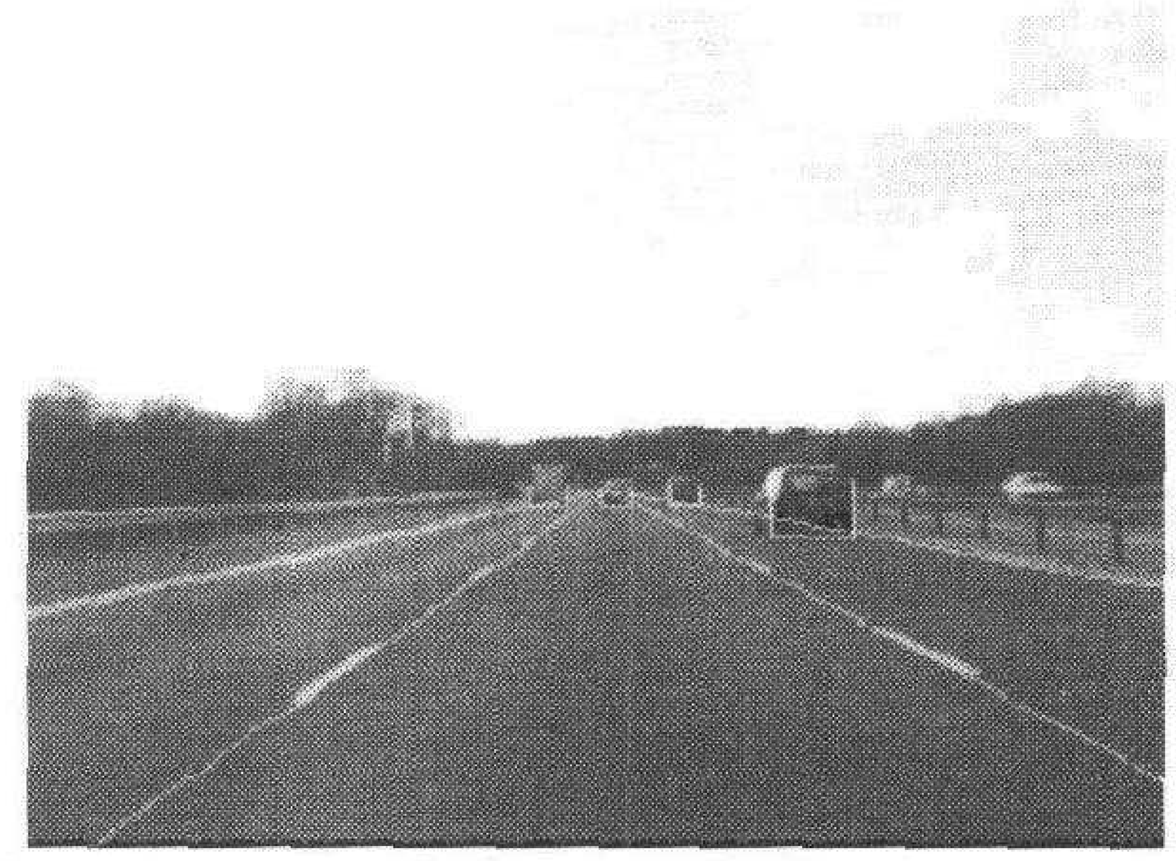


Fig. 8 Outputs of the system at frame 2012. A car is detected in the distance while multiple cars are tracked. The white U shapes are car tracking and detecting results. The white straight lines are lane detection results

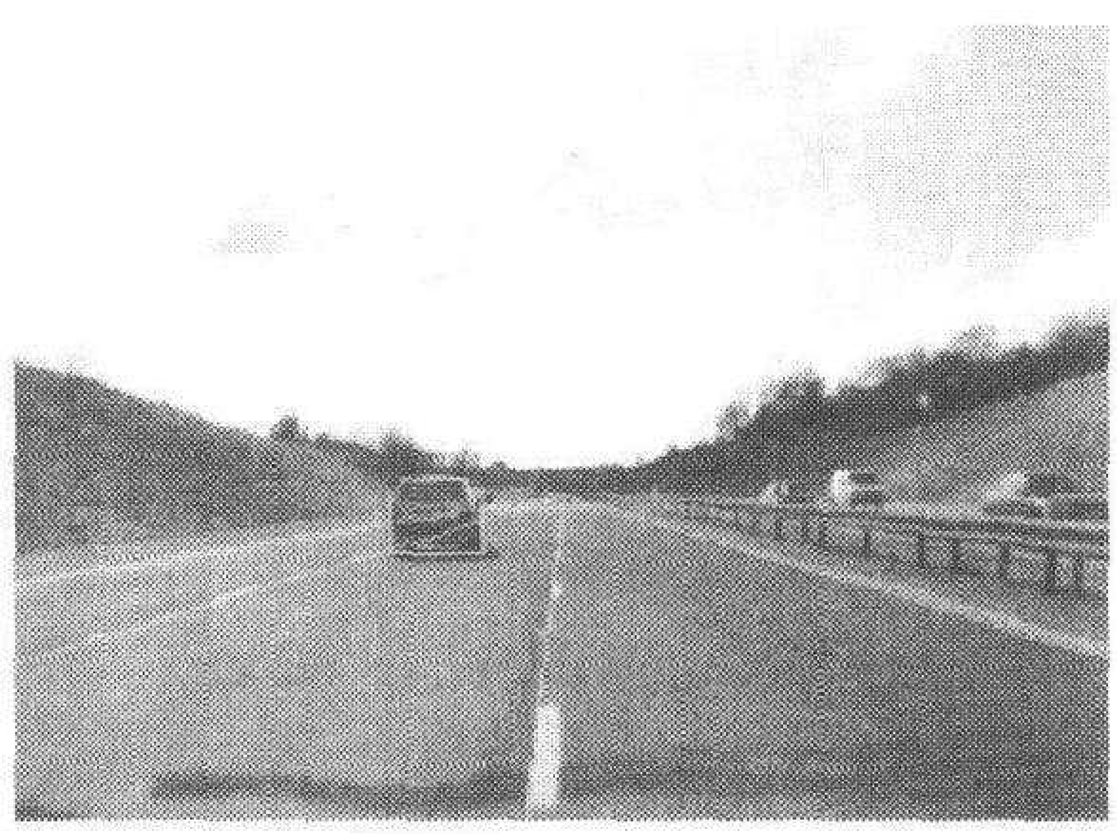


Fig. 9 Outputs of the system at frame 2779. The car is tracked while the camera-installed car is turning right from the middle lane to the right lane. The white U shapes are car tracking and detecting results. The white straight lines are lane detection results

8 Conclusions

This paper presents a vision system for multiple car tracking on the highway. The system includes four main modules: lane detection, car detection, multiple car trackers, and a process coordinator. The important feature of our system is that it uses separate 2-D models for passing car and distant car, and switches them based on the positions of tracked cars. In addition, although our system takes advantage of the prior knowledge of the structure of cars, this knowledge is not as strong as that in [6,9,10]. However, the system needs more tests for improvement, especially under the critical conditions, such as bad weather, inverse lighting and occlusion events.

Acknowledgements I would like to thank Prof. MA Song-De, my adviser for his support through my work. In addition, I am very grateful to Prof. TAN Tie-Niu and HU Zhan-Yi for their comments on the paper.

References

1 Bertozzi M, Broggi A, Fascioli A. Vision-based intelligent vehicles: State of the art and perspectives. *Journal of Robotics and Autonomous Systems*, 2000, **32**(1): 1～16

2 Masaki I Editor. Vision-Based Vehicle Guidance. Berlin: Springer-Verlay, 1992

3 Weber J, Koller D, Luong Q T, Malik J. An integrated stereo-based approach to automatic vehicle guidance. In: Proceedings of ICCV, 1995. 12～20

4 Smith S, Brady J. ASSET-2: Real-time motion segmentation and shape tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1995, **17**(8): 814～829

5 Bertozzi M, Broggi A, Castelluccio S. A real-time oriented system for vehicle detection. *Journal of Systems Architecture*, 1997, **43**(2): 317～325

6 Ferryman J M, Maybank S J, Worrall A D. Visual surveillance for moving vehicles. *International Journal of Com-*

puter Visoin, 2000,**37**(2):187~198

7 Betke M, Harıtaoglu E, Davıs L. Multiple vehicle detection and trackıng in hard real-time. *IEEE Symposıum on Intellıgent Vehicles*, 1996. 351~356

8 Bertozzı M, Broggi A. GOLD: A parallel real-tıme stereo vision system for generic obstacle and lane detection. *IEEE Transactıons on Image Processing*, 1998,**7**(1): 62~81

9 Tan T N, Sullivan G D, Baker K D. Model-based localization and recognition of road vehicle. *International Journal of Computer Visoın*, 1998,**27**(1): 5~25

10 Tan T N, Baker K D. Efficient ımage gradient-based vehıcle locatıon. *IEEE Transactions on Image Processıng*, 2000, **9**(8):1343~1356

11 Press W S, Teukolsky S A, Vetterling W T, Flannery B P. Numerical Recipes in C(2nd Edition). London: Cambridge Unıversity Press, 1992

ZENG Zhi-Hong Received his Ph. D. degree in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, P. R. China in 2002. He is now is a Postdoctor researcher at UIUC, U. S. A.. His research interests include particle filtering, visual tracking, multimode computer-human interactive system .

高速公路中的行车道检测和车辆跟踪

曾智洪

（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: zhzeng@liama. ıa. ac. cn）

摘 要 提出了一种有效的高速公路检测和多车辆跟踪视觉系统. 该系统主要模块包括道路检测、基于二维模型的车辆跟踪器(近距离车辆的矩形模型和远距离车辆的U形模型)、启发式车辆检测、系统协调器. 在系统中，跟踪器的动态产生和终止优化了系统的计算资源. 另外，系统利用鲁棒性估计技术提高了道路检测的性能. 车辆的跟踪是采用三参数状态空间的多边形拟合技术来实现的. 本文采用了PETS2001提供的图像序列测试了系统有效性，在Pentium III 450MHz PC上系统的处理速度为平均每帧12毫秒.

关键词 车道检测，车辆跟踪，边缘点，多边形拟合

中图分类号 TP391.41