# An Efficient Road Detection Method In Noisy Urban Environment

Geng Zhang, Nanning Zheng, Chao Cui, Yuzhen Yan and Zejian Yuan

***Abstract*— Road detection is a crucial part of autonomous driving system. Most of the methods proposed nowadays only achieve reliable results in relatively clean environments. In this paper, we combine edge detection with road area extraction to solve this problem. Our method works well even on noisy campus road whose boundaries are blurred with sidewalks and surface is often covered with unbalanced sunlight. First, segmentation is done and the segments which belong to road are chosen and merged. Second, we use Hough transform and a voting method to get the vanishing point. Then, the boundaries are searched according to the road shape. We also employ prediction to make our method achieve better performance in video sequence. Our method is fast enough to meet real-time requirement. Experiments were carried out on the intelligent vehicle SpringRobot (Fig. 1) on campus roads, which is a good representation of urban environment.**

(a) The testbed vehicle (b) The frontal camera

Fig. 1. (a) is the SpringRobot, which is an autonomous vehicle developed by Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University. It has cameras, lasers and radar. The sensor we use in our experiment is the frontal camera in (b).

## I. INTRODUCTION

Road detection is a key requirement for the vision system of the intelligent vehicle. It is important for the lateral and longitudinal vehicle control, collision avoidance and road following. During the past decades, people have proposed many approaches for road detection based on vision [1]. Most of them were aimed at lateral vehicle guidance on highways or other well-structured roads [2]-[5]. The structured roads usually have clear boundaries and most of them also have man-made markings. In these environments, road detection is simplified to the detection of lines, curves and road markings. For example, the GOLD system at Parma University, Parma, Italy, re-projects the image ahead of the vehicle onto the ground plane and extracts "left-middle-right" groups of road markings [2].

The detection of structured road has got impressive achievements, however, for a large part of the real world environments, there are few markings on the road surface and the boundaries of the road are sometime very fuzzy. So in the recent years, scientists have become enthusiastic over the research of road detection on these unstructured roads. Several methods on this problem have been proposed and plausible results are shown in the literatures [6]-[9]. For example, the SCARF [6] and UNSCARF [7] system at Carnegie Mellon University (CMU), Pittsburgh, PA, have been dealing with unstructured roads in park, where the color of the road is similar to that of the surrounding. But the proposed unstructured road detection algorithms are either lack of robustness to noise or time consuming.

Segmentation has been proved to be a good way to solve unstructured road detection problem. In recent years, various segmentation algorithms have been introduced into road detection. Watershed algorithm [5], region growing [9], texture based segmentation [11] and even more complex methods are used in the literatures.

Campus, as a good representation for urban environment, has always been challenging to the road detection algorithms. Because a lot of campus roads have neither road markings nor clear boundaries, and the surfaces' color distribution is often disturbed by the shadows of the sidewalk trees. To cope with such an environment, road boundary detection should be combined with road area segmentation [8]. Since the former can give us some initial information on the boundaries of road area and the latter can distinguish the road area from the surroundings based on feature homogeneity. To make a good detection and meet the real-time requirement, efficient segmentation algorithm need to be employed and a proper framework of boundary detection should be build.

In this paper, we employ a fast segmentation algorithm [10] to detect the road area. This segmentation method is a pixel fusion procedure and various informations can be used to achieve best performance. In our method, considering the real-time requirement, color information is chosen. To get better boundaries of the road, vanishing point detection is also integrated in our method. The whole detection system works as the block diagram in Fig. 2. With this method, we

This research is supported in part by the National Basic Research Program of China under Grant No.2007CB311005, the National Natural Science Foundation of China under Grant No.60635050.

Geng Zhang is with Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, Shaanxi, China zhangtetsu@gmail.com

Nanning Zheng is an IEEE Fellow, he is now with Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, Shaanxi, China nnzheng@aiar.xjtu.edu.cn

Chao Cui is with Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, Shaanxi, China stevens.cui@gmail.com

Yuzhen Yan is with Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, Shaanxi, China yyz.yanyuzhen@stu.xjtu.edu.cn

Zejian Yuan is with Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, Shaanxi, China yzejian@gmail.com

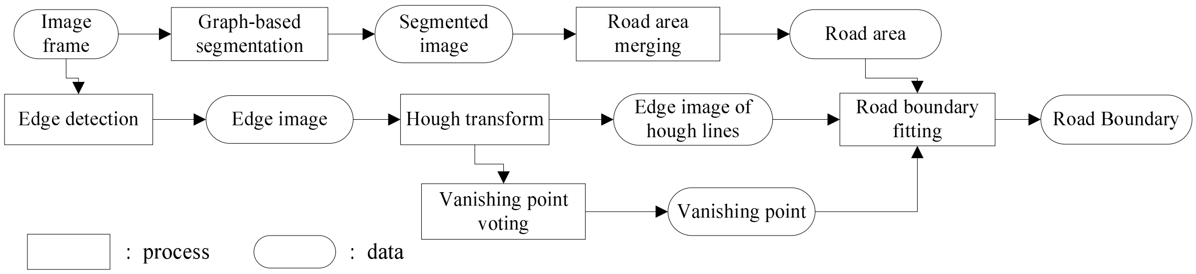



Fig. 2. The block diagram of our method

process more than 20 frames per second.

This article is organized as follows: section II introduce the graph-based segmentation we use and explain how to choose the road segments. In section III we combine road area segmentation with vanishing point detection to get the road boundary. Section IV includes the experiments and some results. Finally, we discuss the future work and make the conclusion in section V.

## II. ROAD AREA DETECTION

The campus road surface is usually covered with shadows of the sidewalk trees, which greatly affect the consistency of the appearance feature, for example, color and texture. Another difficulty is that some edges that the shadows form can be stronger than that of the road boundaries. So any attempts which want to separate the whole road area will lead to failure in this lighting condition. To deal with this, first we over-segment the image into pieces. Considering that the road boundary is always locally distinctive, we employ a graph-based segmentation which is good at capturing the local details. This algorithm is proposed by Pedro F. Felzenszwalb in [10].

The segments of the image can by classified into two classes: road and non-road. All of them have several features including color, position and size. We use these features to separate the two classes and merge the small segments into the whole road area.

### A. Graph-Based Segmentation

Some graph-based segmentation algorithms have been proposed many years ago. In 1971, Zahn presented a segmentation method using minimum spanning tree (MST). But the former approaches are easily affected by the difference of pixels in the high variability region, so they are not suitable to segment the noisy road image. The segmentation algorithm we use has the characteristic of preserving detail in low-variability image regions while ignoring detail in high-variability regions.

In the graph-base algorithm, an image is represented as an undirected graph $G = (V, E)$. The vertices $v \in V$ represent the pixels or segments, and each edge $(v_i, v_j) \in E$ has a weight $w(v_i, v_j)$. The weight of the edge measures the similarity of the two segments connected by it. At the beginning, each vertical represent a pixel and the edges are constructed in an 8-connected sense. For a gray-scale image, we can simply define the weight based on the absolute difference between pixels,

$$w(v_i, v_j) = |I(p_i) - I(p_j)|, \tag{1}$$

where $I(p_i)$ is the intensity of pixel $p_i$. In our system, color information is used, so we compute the similar weight measure to each channel of the image. That is, we do the algorithm to the red, green and blue planes respectively and intersect these three sets.

The segmentation is a merging procedure during which the components (pixels or segments) are merged according to their internal difference and the difference between them. The internal difference is defined as the largest weight in the MST of the component:

$$Int(C) = \max_{e \in MST(C,E)} w(e), \tag{2}$$

where $C \in V$ represents one of the components. And the difference between two components $C_1, C_2 \in V$ is the minimum weight edge connecting them:

$$Dif(C_1, C_2) = \min_{v_i \in C_1, v_j \in C_2, (v_i, v_j) \in E} w(v_i, v_j) \tag{3}$$

During the segmentation, we merge the two disconnected component as long as the difference between them is smaller than their minimum internal difference $MInt$, which is defined as follows:

$$\begin{aligned} &MInt(C_1, C_2) \\ &\quad = min(Int(C_1) + \tau(C_1), Int(C_2) + \tau(C_2)), \end{aligned} \tag{4}$$

In this definition, the threshold function $\tau$ controls the comparing degree. To avoid the affection caused by the relatively strong edges in the cluttered area, $\tau(C)$ is defined to be $k/|C|$, where $k$ is a constant and $|C|$ represents the size of component $C$. This definition ensures that for small components we require stronger evidence for a boundary. More detail of this algorithm can be found in [10].

To make sure that this algorithm is suitable for our road detection system. We modify the function $\tau$ according to some prior knowledge. First, when the vehicle is running, the position of the camera is near the center of the road. So the main part of the road area tends to be in the middle of the image. Second, when the road is projected to the image coordinates, the size of the road region increases as it comes to the bottom of the image. So it is reasonable that the regions which is closer to the vertical middle line and bottom of the image are more likely to be the road area and should be merged into one.

Considering the two constrain above, we redefine $\tau$ as

$$\tau(C) = \frac{ke^{\frac{2d_{mid}}{w} + \frac{d_{bottom}}{h}}}{|C|}, \quad (5)$$

where $w$ and $h$ are the width and height of the image. $d_{mid}$ is the minimum distance between the segment $C$ and the vertical middle line. $d_{bottom}$ is the minimum distance between the segment $C$ and the bottom of the image. More formally, we define the two distances as

$$d_{mid} = \min_{p \in C} |p_x - \frac{w}{2}|, \quad (6)$$

$$d_{bottom} = \min_{p \in C} |p_y - h|, \quad (7)$$

where $p$ represent a pixel in the component C and the coordinate of $p$ is $(p_x, p_y)$.

We segment the image using this algorithm. The result of segmentation could be seen in Fig. 3 where the fuzzy road boundaries have been well captured despite of the shadow and other noise. The road segments can then be classified and merged.

### B. Road Area Merging

To get the whole road area or so called drivable region, we use multiple cues to classify the segments. The segments are clustered into two classes: road and non-road.

First, size and position of the segments are considered to be important for classification. As we see in the result above, despite that there is noise on the road surface, the main part of the road is tend to be merged into one or two segments (sometimes separated by the middle line of the roadway). And another important prior knowledge is that the segment which attaches the bottom of the image is surely to be part of the road area, as long as the vehicle is running on the road. During the process, the bumper of the vehicle is cut out of the image.

Some road segments which are near the top of the image may be separated from the main part of the road area. We use color information to find these segments. Although the changing of the global illumination usually makes the color of the road surface different between frame in video sequence, the statistical histograms of the road segments' color is consistent within one frame. So we suppose that in one image, the selected biggest road segment represents the color characteristic of the road area. Color histogram is used in our method, which is computed in RGB space (HSV is also adoptable). For a segment, we compute its color histogram by discretizing every channel into 64 scale levels and connect the histograms of the 3 channels together to form the segment's color histogram (SCH).

When the biggest part is got, we compute SCHs for other segments which are near the selected parts and not too small to be a road segment. $\chi^2$ distance is used to measure the similarity between histograms.

$$\chi^2(SCH, SCH_1) = \frac{1}{2}\sum_i \frac{(SCH^i - SCH_1^i)^2}{(SCH^i + SCH_1^i)}, \quad (8)$$

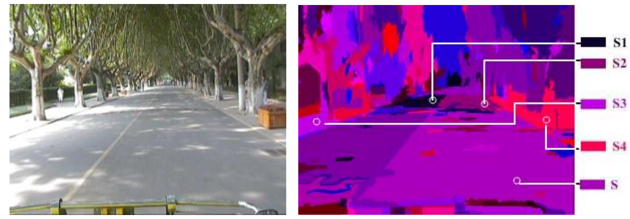



Fig. 3. The result of the segmentation

where $SCH$ belongs to the biggest road segment, and $SCH_1$ belongs to one of the candidate segments. $SCH^i$ represents the i-th bin in $SCH$. We set a threshold value $\epsilon$, if

$$\chi^2(SCH, SCH_1) < \epsilon, \quad (9)$$

the segment of $SCH_1$ will be selected.

In Fig. 3, the biggest part is segment $S$, the two road segments to be selected are $S1$ and $S2$. Contrastively, the two segments which are not belong to road area are $S3$ and $S4$. Fig. 4 shows the SCHs of these segments and the similarity of the other SCHs compared with that of the biggest segment.

After merging the main segments, we employ a dilation algorithm on the binary image of the road area to fill the hollows which are caused by a few very strong shadows and some small landmarks. But we just use the dilation to fill the very small hollows to make the road area more complete. We don't fill the big holes because these hollows do not affect the boundary fitting too much, and we want the "blank space" caused by people and other vehicles not be considered as road. That is to say, the road area we detect do not cover the obstacles on it.

In the next section, we will use this detected result and the shape information of the road to find the boundaries.

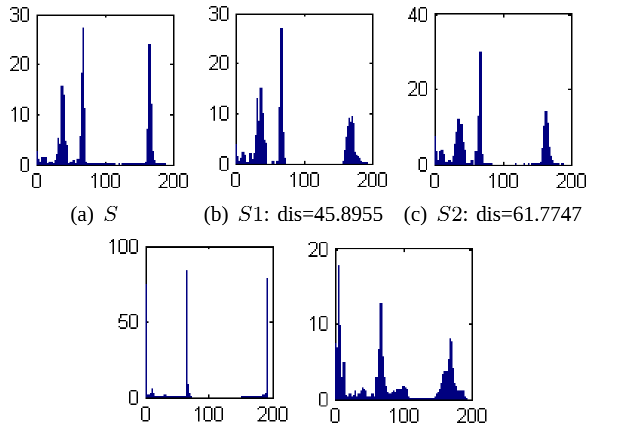
## III. DETECTING ROAD BOUNDARIES USING VANISHING POINT

Although we can get a satisfying result of the road area using the method above, it is still hard for the intelligent vehicle to understand the structure of the road. So finding the vanishing point and road boundaries is also important.

### A. Vanishing Point Detection

Hough transform is a traditional but efficient way to detect the vanishing point [12], especially when the road is nearly straight. The procedure is done on the gray-scale image. For a structured road image, in which the roadway lines are clear and the color of the road is consistence, vanishing point can be easily detect by extending the lines got from Hough transform. But if the road is unstructured, especially the campus road in our experiment, the line segments detected by Hough transform are cluttered. We can not get the vanishing point from them directly even by using voting algorithm [13].

We should detect as many Hough lines passing the vanishing point as possible so that the voting algorithm can by used. According to the appearance of the images, the lower part which is closer to the vehicle is relatively clean. So we do edge detection and Hough transform to part of the image.



(a) $S$ (b) $S1$: dis=45.8955 (c) $S2$: dis=61.7747

(d) $S3$: dis=237.6672 (e) $S4$: dis=138.8290

Fig. 4. The SCHs of the marked segments in Fig. 3. (a) is belongs to the biggest road segment $S$; (b) and (c) belong to the left and right road segments $S1$ and $S2$; (d) and (e) belongs to the left and right non-road segments $S3$ and $S4$. The numbers below each SCH represents its $\chi^2$ distance to the standard SCH of $S$.

Before detecting edges with canny operator, we cut the upper part of the image. Then canny detection and Hough transform is done to it. We also use a constraint to choose the line we need. The line segments which are nearly horizontal are excluded. More formally, a line segment $\overline{AB}$ is kept when

$$|\frac{A_y - B_y}{A_x - B_x}| > \delta, \tag{10}$$

where $\delta$ is a threshold value which we set to be $\tan(30^o)$.

The position of the horizontal line is also important. When on the flat ground plane and the camera's extrinsic parameters are set, the position of this line in the image is fixed. As in Fig. 5, the vanishing points is always near the horizontal line. The region where the vanishing point might be in is called the vanishing belt. So in the voting procedure, we only computed the number of votes for the points which are in the vanishing belt.

Given $\mathbf{L}$ is the set of the selected hough lines $\mathbf{L} = \{(a^1, b^1), (a^2, b^2), \cdots, (a^n, b^n)\}$, where $a, b$ are the head and end points of a line. We vote a candidate point $p$ in the vanishing belt if

$$|\frac{p_x - a_x^i}{p_y - a_y^i} - \frac{p_x - b_x^i}{p_y - b_y^i}| < \alpha, \tag{11}$$

where $p_x$ and $p_y$ represent the x and y coordinate of point $p$. After the voting process, the points which have the most votes are selected and the average of them is the vanishing point.

In the video sequence, we assume that the position of the vanishing point do not change much between coherent frames. So we just vote for points which are near the last vanishing point in the vanishing belt.

### B. Boundary Fitting

Road model is often used to help detect the boundaries of unstructured road [14]. For the road which is nearly straight, the shape of it can be approximate to the model in Fig. 5.

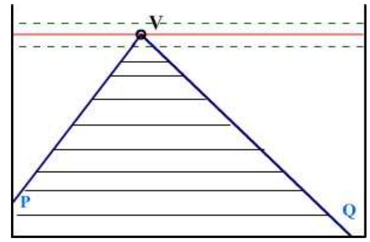



Fig. 5. The model of the road

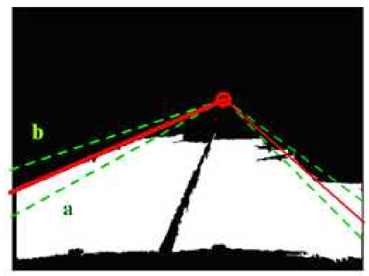



Fig. 6. The red lines are the candidate road boundaries, and the green dotted lines are the assistant lines.

The vertex $V$ of the model is the vanishing point detected above. So we just need to get the positions of the other ending points $P$ and $Q$.

Points $P$ and $Q$ glide along the boarder of the image around the bottom left and bottom right corner. So we assume that these two points move in the following range,

$$\begin{aligned} P \in \{&(x_p = 0, y_p \in [h - \delta h, h]) \cup \\ &(x_p \in [0, \delta w], y_p = h)\}, \\ Q \in \{&(x_q = w, y_q \in [h - \delta h, h]) \cup \\ &(x_q \in [w - \delta w, w], y_q = h)\}, \end{aligned} \tag{12}$$

where $w$ and $h$ are the width and height of the image and the origin is at the upper-left corner.

We search $P$ and $Q$ in the range above to find the road boundaries $\overline{VP}$ and $\overline{VQ}$. As in Fig. 6, to judge whether a line fits the road boundary, the ratio of the road area between both sides of this line is used as a measurement. We compute the length of the two assistant lines that fall into the road area (the effective length). The ratio $r$ can be written as

$$r = \frac{l_a}{l_b + \delta}, \tag{13}$$

where $l_a$ and $l_b$ are the effective length of the two assistant lines and $\delta$ is a very small constant to avoid division by zero (like the situation shown in Fig. 6). We consider the line which has the biggest $r$ to be the boundary of the road.

## IV. EXPERIMENTS

Our experiments were taken on the SpringRobot, which is an autonomous vehicle developed in Xi'an Jiaotong University, Xi'an, China. The experiments demonstrate the efficiency and robustness of our method. The method proposed above has been implemented in C++ on a dual core 3GHz PC with memory of 2GB. We did all the experiments on the video sequences with resolution of 320×240, which were taken on the campus roads of Xi'an Jiaotong University.

We use the graph-based segmentation algorithm to aggregate the road and non-road areas in to pieces. The original code of the segmentation is provided by Pedro F. Felzenszwalb. In this procedure, a Gaussian smoothing is used, the variance of the Gaussian kernel should not be set too big in case the weak boundaries of road are blurred. The constant $k$ in (5) controls the size of the segment pieces, which is usually set as 200-300 in our experiments. After selecting and merging the road segments, we get the whole road area.

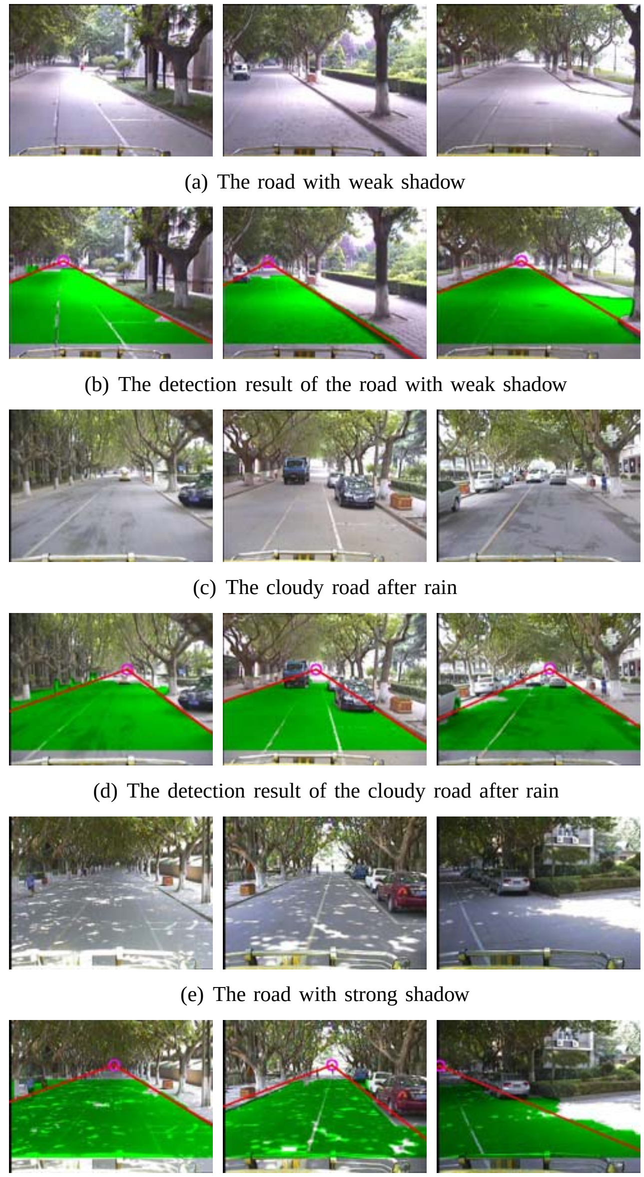
When detecting the vanishing point, first, position of the horizonal line is predetermined according to the size of image and the external parameters got from the camera calibration. Then, the Hough transformation function in the OpenCV package is used to extract line features.

Fig. 7 shows some of detection results on the normal straight roads in campus. Normally, we can get good approximation of the road boundary even if the road area is not accurate in some conditions. From the results, we see that our method can solve the problem of shadow, high-lighting and water track. The road model we use adapts to various road orientations as long as the vanishing point can be detected. Our detection method is based on road area segmentation, so when either side of the road boundary is covered by the parked cars, our detected area will be at the inner side of them. This makes sense because we do not expect the autonomous vehicle to run into that area.

Some times the shadows on the road is so strong that the segmentation algorithm can not merge them into one piece. As shown in Fig. 8, the segments of shadow or high-lighting area far from the vehicle are hard to be classified as road area. But our vanishing point detection algorithm seldom fails, except in the second scene where the shadow area is too big and the right side of the road is an open crossing that little useful lines can be detected by Hough transform. For most of the time, we can get good result of the road boundaries despite that the whole area is not completely detected.

The average detection speed of our method is approximately 50 millisecond per image ($320 \times 240$). So that we can process 20 frames per second, which is enough to achieve the real-time requirement. For the video sequence, prediction is used to assure the continuity between frames. In our experiment, we use prediction when detecting vanishing point and fitting road boundaries to avoid the result from changing suddenly. Some frames extracted from a video sequence are shown in Fig. 9.

Fig. 10 shows some of the detection of corner scenes, where vanishing point do not exist. So we simply merge the road segments. We can see the road area detection results are good in these conditions.



(a) The road with weak shadow

(b) The detection result of the road with weak shadow

(c) The cloudy road after rain

(d) The detection result of the cloudy road after rain

(e) The road with strong shadow

(f) The detection result of the road with strong shadow

Fig. 7. Some detection results under various conditions

## V. CONCLUSIONS AND FUTURE WORKS

### A. Conclusion

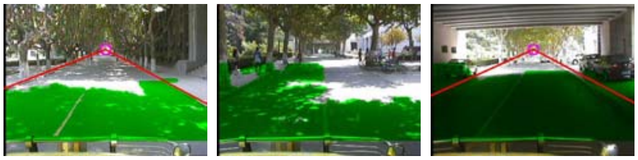
Unstructured road detection is a challenging problem in autonomous vehicle vision system. We use a graph-based segmentation method to detect road area. Then we detect vanishing point with Hough transformation. Finally, an adaptive road model is used to detect road boundaries. Prediction is used when detecting road in video sequence so that the continuity between frames is assured. The most important advantage of our method is its speed. Real-time requirement can be totally fulfilled.

From the results we can see, for most of the road conditions our method works well. For when the shadow or high-lighting area is big and strong, the area detection result may not be good, but the boundary detection does not fail easily.

### B. Future Works

Although our method provides good results in various conditions, it may fail when the surface of the road is

(a) The road with very strong shadow



(b) The detection result of the road with very strong shadow

Fig. 8. Detection results under extreme conditions

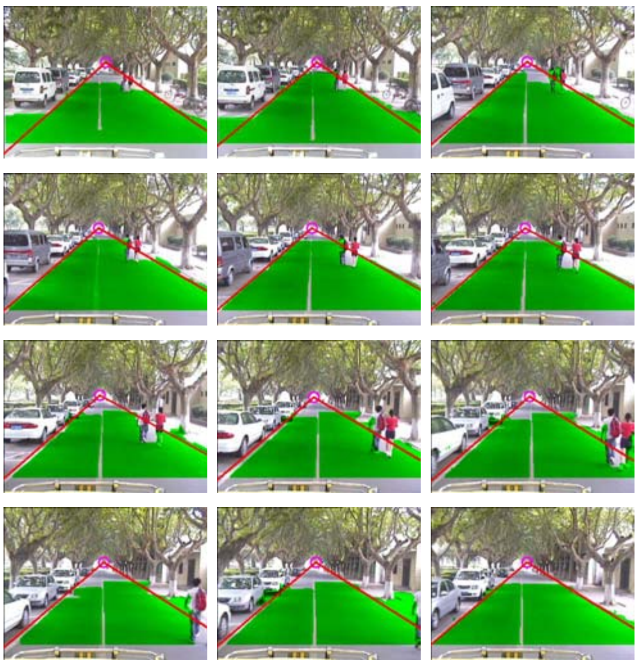


Fig. 9. The detection results extracted from a video sequence

(a) The road of the corner scenes

(b) The detection results of the corner scenes

Fig. 10. Road area detection at corners

too cluttered. Preprocessing such as shadow weakening is needed to achieve better performance. And prediction can be used to road area detection. For example, correlation of the corresponding pixels between frames can be constructed so that the sudden changing of the road surface will not easily affect the road area segmentation and merging.

## VI. ACKNOWLEDGEMENT

Thanks to professor Pedro. F. Felzenszwalb, who provides the graph-based segmentation package.

## REFERENCES

[1] M. Bertozzi, A. Broggi and A. Fascioli, "Vision-based intelligent vehicles: State of the art and perspectives," *Robotics and Autonomous Systems*, 32, 2000, pp. 1-16.

[2] M. Bertozzi and A. Broggi, "GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection," *IEEE Transactions On Image Processing*, Vol.7, No.1, January 1998.

[3] D. Pomerleau, "RALPH: Rapidly adapting lateral position handler," *in Proc. IEEE Intelligent Vehicles Symp.*, Detroit, MI, Sept. 1995, pp. 506-511.

[4] P. Katsande and P. Liatsis, "Adaptive order explicit polynomials for road edge tracking," *in Proc. IEEE Int. Conf. Advanced Driver Assistance Systems, (ADAS01)*, Sept. 2001, pp. 63-67.

[5] S. Beucher and M. Bilodeau, "Road segmentation and obstacle detection by a fast watershed transform," *in Proc. IEEE Intelligent Vehicles Symp.*, October 1994, pp. 296-301.

[6] J. D. Crisman and C. E. Thorpe, "SCARF: A color vision system that tracks roads and intersections," *IEEE Trans. Robot. Automat.*, vol. 9, Feb. 1993, pp. 49-58.

[7] J. Crisman and C. E. Thorpe, "UNSCARF: A color vision system for the detection of unstructure roads," *in Proc. IEEE Int. Conf. Robotics and Automation*, Apr. 1991, pp. 2496-2501.

[8] Y. He, H. Wang, and B. Zhang, "Color-Based Road Detection in Urban Traffic Scenes," *IEEE Transactions On Intelligent Transportation Systems*, Vol. 5, No. 4, December 2004, pp. 309-318.

[9] J. M. Alvarez, A.M. Lopez, R. Baldrich, "Shadow resistant road segmentation from a mobile monocular system," *Pattern Recognition and Image Analysis*, Springer Berlin/ Heidelberg, Volume 4478, 2007, pp. 9-16.

[10] P. F. Felzenszwalb and D. F. Huttenlocher, "Efficient Graph-Based Image Segmentation," *International Journal of Computer Vision*, 59(2), 2004, pp. 167-181.

[11] J. Zhang and H. Nagel, "Texture-based segmentation of road images," *in Proc. IEEE Intelligent Vehicles Symp.*, Oct. 1994, pp. 260-265.

[12] B. Yu and A. K. Jain, "Lane Boundary Detection Using A Multiresolution Hough Transform," *on Proc. Image Processing*, vo. 2, Oct. 1997, pp. 748-751.

[13] C. Rasmussen, "Texture-based vanishing point voting for road shape estimation," *in Proc. British Machine Vision Conf.*, Sept. 2004.

[14] S. Richter and D. Wetzel, "A robust and stable road model," *In Proc. of 12th International Conferences on Pattern Recognition*, vol. 1, Oct. 1994, pp. 777-780.