Recognizing white line markings for vision-guided vehicle navigation by fuzzy reasoning [1]

Wei Li *, Goutao Lu, Yongqiang Wang

Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China

Abstract

This paper presents a new method for recognizing white line markings for navigation of an autonomous vehicle that operates in out-door environments. A fuzzy-reasoning-based general technique for edge detection is used to classify a pixel in an image into a border region or a uniform region based on luminance differences between this pixel and its neighboring pixels. The idea of this paper is to integrate special knowledge about white line markings into a fuzzy rule base using this general technique. The method is implemented on the THMR-III mobile robot, and experiments show that this method is robust to noise and environment changes. © 1997 Elsevier Science B.V.

Keywords: Image processing; Fuzzy reasoning; Robot vision; Navigation; Autonomous vehicle

1. Introduction

Detection of white line markings on streets is one of the most important steps in robot motion (Graef and Kuhnert, 1988, Thorpe et al., 1991). However, it is not easy to achieve this task because of the vagaries of natural illumination, bright sunlight, clouds, and so forth. Several approaches have been proposed for detection of edges in an image, such as, the evaluation of an estimate of the local gray level gradient or of the local gray level second derivative (Jain, 1989). These approaches, however, might not be feasible for robot navigation in out-door environments since such a robot requires image processing in real-time, and robustness to noise and environ-

ment changes. To speed up image processing, a simple thresholding algorithm based on statistics is implemented on the THMR-III mobile robot (Yao, 1994). In experiments, it is noticed that thresholds for edge extraction depend largely on environment changes. Besides, different thresholds should be determined to process different regions in the same image to obtain the desired result.

Recently, some approaches to edge detection based on computational intelligence have been proposed. For instance, some approaches present algorithms for extracting edges in an image using neural networks (Moura and Martin, 1991, Lepage and Poussart, 1992). Since out-door environments usually are very complicated, providing complete and efficient patterns with neural networks is very difficult. Also, some researchers propose general purpose methods for edge detection using fuzzy reasoning (Tao et al., 1993, Russo and Ramponi, 1994, Bezdek

---

and Shirvaikar, 1994, Bezdek et al., 1995). They use fuzzy reasoning to classify a pixel in an image into a border region or a uniform region based on luminance differences between this pixel and its neighboring pixels.

This paper presents a method for vision based navigation of an autonomous vehicle using fuzzy inference. This study integrates some special knowledge into a fuzzy rule base, using a general approach to edge extraction via fuzzy inference (Russo and Ramponi, 1994). As a result, white line markings on streets or on roads can be efficiently recognized for robot motion control. The method is implemented on the THMR-III mobile robot, and some experimental operations are performed to demonstrate its effectiveness and robustness.

2. Vision system of the THMR-III robot

The THMR-III mobile robot is equipped with a vision system for image processing, which consists of a high-level sub-system and a low-level sub-system, as shown in Fig. 1. The high-level vision sub-system has a pipeline, an image box, and a SUN workstation; while the low-level vision sub-system has a PC586 computer, a high-speed image processing board VIGP-2M, and two CCD cameras.
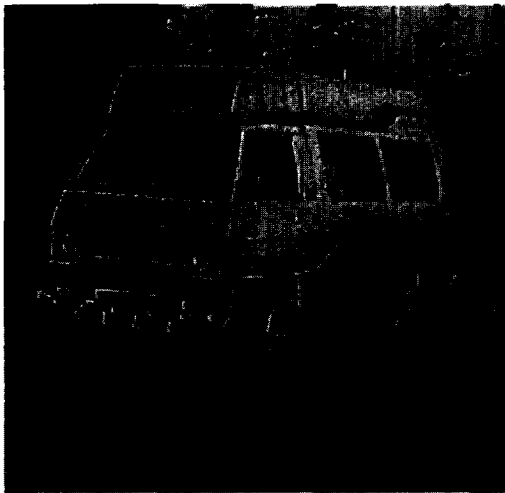

Fig. 1. THMR-III out-door mobile robot.

The CCD cameras are installed on the top and in the front of THMR-III robot according to the geometry of the car body and navigation requirements, respectively. The distance from the top camera to the ground is about two meters; while the distance from the front camera is approximately one meter. Image information, taken by the top camera, is mainly used to recognize edges on roads or white lines on streets and to track them within a region 8 m ~ infinity. Image information, taken by the front camera, is mainly used to avoid obstacles and to track edges or white lines within a region 4 ~ 12 m. All images that are used in this paper are taken by the front CCD camera, for example, the images in Fig. 6(a) and Fig. 7(a).

3. Edge detection by fuzzy reasoning

Fuzzy logic inference is based on the theory of fuzzy sets, as introduced by Zadeh (1965). A fuzzy set A in a universe of discourse X is defined by its membership function $\mu_A(x)$. For each $x \in X$, there exists a value $\mu_A(x) \in [0,1]$ representing the degree of membership of x in X. In fuzzy logic control membership functions, assigned to linguistic variables, are used to fuzzify physical quantities. Fuzzified inputs are fed to a fuzzy rule base. This rule base is used to characterize the relationship between fuzzy inputs and fuzzy outputs. For example, a simple fuzzy inference rule relating input v to output u might be expressed in the *condition–action* form as follows:

If v is W Then u is Y

where W and Y are fuzzy values defined on the universes of v and u, respectively. The response of each fuzzy rule is weighted according to the degree of membership of its input conditions. The inference engine provides a set of control actions according to fuzzified inputs. Since the response actions are fuzzy, a defuzzification method is required to transform fuzzy response actions into a crisp output value of the fuzzy logic inference. A widely used defuzzification method is the centroid method.

Russo and Ramponi (1994) presented an idea of detecting edges by fuzzy inference that classifies

every pixel in an image into white or black based on the following two strategies:

- *if a pixel belongs to a border region then make it black else make it white*
- *if a pixel belongs to a uniform region then make it white else make it black*

In order to decide if a pixel belongs to a border region or to a uniform region, luminance differences between the pixel and its neighboring pixels have to be computed, as shown in Fig. 4. Then, luminance differences are processed by fuzzy reasoning to determine the luminance of this pixel. The fuzzy inference algorithm consists of the fuzzy sets and a rule base. Fig. 2(a) shows the membership functions of inputs. As a crisp number, the luminance differences between the pixel and its neighboring pixels, denoted by $Dif\_Lum(i)$, could be a positive or negative value in $[-255,255]$. For fuzzy reasoning, the luminance differences, $Dif\_Lum(i)$, must be fuzzified by using the membership functions of luminance differences. In Russo and Ramponi (1994), the membership functions of the inputs are defined by triangular functions with peaks at $Dif\_Lum(i) = L_0$ and $Dif\_Lum(i) = -L_0$, as shown by the dotted lines in Fig. 2(a). Using these types of membership functions, the max-
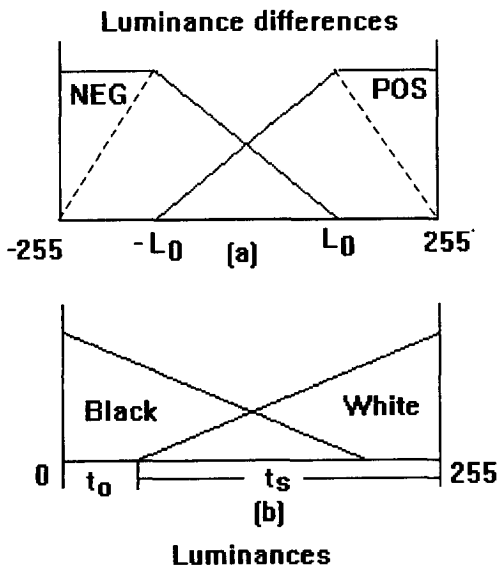

Fig. 3. Rule base of fuzzy logic inference.

Fig. 2. Membership functions of inputs (luminance differences) and outputs (luminances).

imum degree $\mu(L_0) = \mu(-L_0) = 1$ is obtained at $Dif\_Lum(i) = L_0$ and $Dif\_Lum(i) = -L_0$. In other words, the luminance difference $Dif\_Lum(i) = L_0$ (or $Dif\_Lum(i) = -L_0$) contributes the strongest weight to the white color (or the black color). In our study, we noted that, instead of the luminance difference equal to L_0 (or $-L_0$), the luminance differences being larger than L_0 (or smaller than $-L_0$) do contribute the strongest weight to the white color (or the black color). Therefore, we replace the triangular membership functions by the trapezoidal membership functions, as shown by the solid line in Fig. 2(a). The value L_0 is determined by the average value of all luminance differences. Fig. 2(b) shows the membership functions of outputs based on which the gray level intensity of the pixel, $Lum(Q)$, is *determined* by the fuzzy rules, shown in Fig. 3. All these fuzzy rules are given as follows:

Rule 1: If $Dif\_Lum(1)$ *is NEG* and $Dif\_Lum(2)$ *is NEG* and $Dif\_Lum(8)$ *is NEG* and $Dif\_Lum(4)$ *is POS* and $Dif\_Lum(5)$ *is POS* and $Dif\_Lum(6)$ *is POS* Then $Lum(Q)$ *is BLACK*

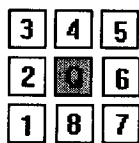Rule 2: If $Dif\_Lum(2)$ *is NEG* and $Dif\_Lum(3)$ *is NEG* and $Dif\_Lum(4)$ *is NEG* and

Fig. 4. Neighbouring pixels of the pixel Q.

Dif_Lum(6) *is POS* and *Dif_Lum*(7) *is POS* and *Dif_Lum*(8) *is POS* Then *Lum*(Q) *is BLACK*

Rule 3: If *Dif_Lum*(6) *is NEG* and *Dif_Lum*(7) *is NEG* and *Dif_Lum*(8) *is NEG* and *Dif_Lum*(2) *is POS* and *Dif_Lum*(3) *is POS* and *Dif_Lum*(4) *is POS* Then *Lum*(Q) *is BLACK*

Rule 4: If *Dif_Lum*(1) *is NEG* and *Dif_Lum*(7) *is NEG* and *Dif_Lum*(8) *is NEG* and *Dif_Lum*(3) *is POS* and *Dif_Lum*(4) *is POS* and *Dif_Lum*(5) *is POS* Then *Lum*(Q) *is BLACK*

Rule 5: If *Dif_Lum*(4) *is NEG* and *Dif_Lum*(5) *is NEG* and *Dif_Lum*(6) *is NEG* and *Dif_Lum*(1) *is POS* and *Dif_Lum*(2) *is POS* and *Dif_Lum*(8) *is POS* Then *Lum*(Q) *is BLACK*

Rule 6: If *Dif_Lum*(1) *is NEG* and *Dif_Lum*(2) *is NEG* and *Dif_Lum*(3) *is NEG* and *Dif_Lum*(5) *is POS* and *Dif_Lum*(6) *is POS* and *Dif_Lum*(7) *is POS* Then *Lum*(Q) *is BLACK*

Rule 7: If *Dif_Lum*(3) *is NEG* and *Dif_Lum*(4) *is NEG* and *Dif_Lum*(5) *is NEG* and *Dif_Lum*(1) *is POS* and *Dif_Lum*(7) *is POS* and *Dif_Lum*(8) *is POS* Then *Lum*(Q) *is BLACK*

Rule 8: If *Dif_Lum*(5) *is NEG* and *Dif_Lum*(6) *is NEG* and *Dif_Lum*(7) *is NEG* and *Dif_Lum*(1) *is POS* and *Dif_Lum*(2) *is POS* and *Dif_Lum*(3) *is POS* Then *Lum*(Q) *is BLACK*

Fig. 4 shows a pixel, Q, and its neighboring pixels. Fig. 5 illustrates how to compute the black color strength of the pixel, w_t, using *Rule 1*. First, the luminance differences between the pixel with its neighboring pixels 1, 2, 4, 5, 6, and 8, are computed and are denoted by, $Lum\_Dif(1)$, $Lum\_Dif(2)$, $Lum\_Dif(4)$, $Lum\_Dif(5)$, $Lum\_Dif(6)$, and $Lum\_Dif(8)$. Then, $Lum\_Dif(1)$, $Lum\_Dif(2)$, and $Lum\_Dif(8)$ are fuzzified using the membership functions, NEG (negative), to get their membership
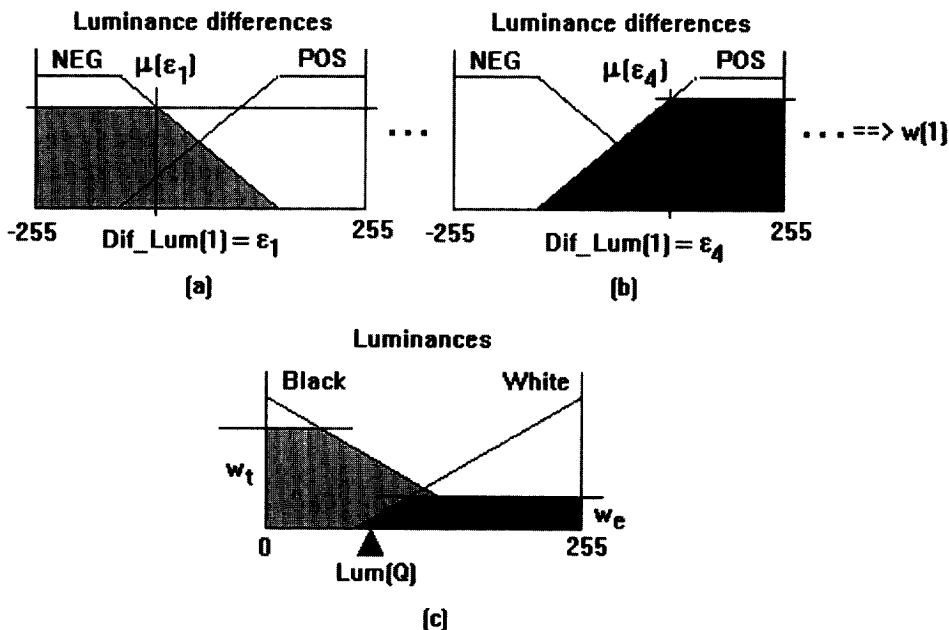

Fig. 5. Image processing by fuzzy reasoning.

degrees $\mu(1)$, $\mu(2)$, $\mu(8)$, respectively, shown in Fig. 5(a); while $Lum\_Dif(4)$, $Lum\_Dif(5)$, $Lum\_Dif(6)$ are fuzzified by the membership functions, POS (positive), $\mu(4)$, $\mu(5)$, $\mu(6)$, respectively, shown in Fig. 5(b). The black color strength of *Rule 1*, $w(1)$, is computed as follows:

$$w(1) = \frac{\mu(1) + \mu(2) + \mu(8) + \mu(4) + \mu(5) + \mu(6)}{6}.$$

(1)

Obviously, the more negative all $Lum\_Dif(1)$, $Lum\_Dif(2)$, and $Lum\_Dif(8)$ are and the more positive all $Lum\_Dif(4)$, $Lum\_Dif(5)$, $Lum\_Dif(6)$ are, the bigger the weight value, $w(1)$, of the black color is. For each rule, i ($i = 1,\ldots,8$), the fuzzy reasoning outputs its correspond weight $w(i)$. The last step, the luminance of the pixel is computed by centroid defuzzification based on $w(i)$ ($i = 1,\ldots,8$), shown in Fig. 5(c), and this is expressed by the following equations,

$$w_t = \max\{w(i)\},$$

(2)

$$w_e = 1 - w_t,$$

(3)

$$Lum(i) =$$
$$\frac{w_t + 2.0 \times w_t \times w_e + w_e}{0.333 \times t_o - 0.6667 \times w_t^3 \times t_o + w_t \times w_e \times t_s + w_t \times (t_o + t_s)},$$

(4)

where w_e is the weight of the white color produced by the rule base, $Lum(i)$ is the luminance of the pixel, i.e., the gray level intensity, and t_o and t_s are the parameters of the membership functions in Fig. 2(b) and are determined purely empirically. Fig. 6(b)Fig. 7(b) show 512×512 pixel images that are processed by this algorithm.

4. Robot navigation based road knowledge

The image process board VIGP-2M provides an 512×512 pixel image. In order to speed up the image processing, such an image is first compressed into an 128×128 pixel image. Then, all pixels in the image are classified into a border region or a uniform region by fuzzy logic inference. In this step, noise in an image can be filtered to detect edges
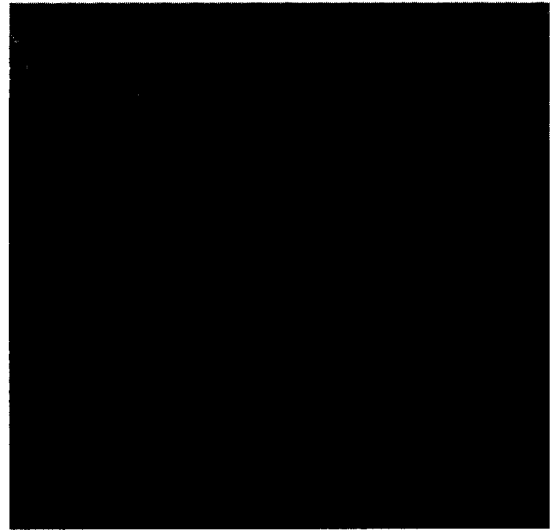

(a)
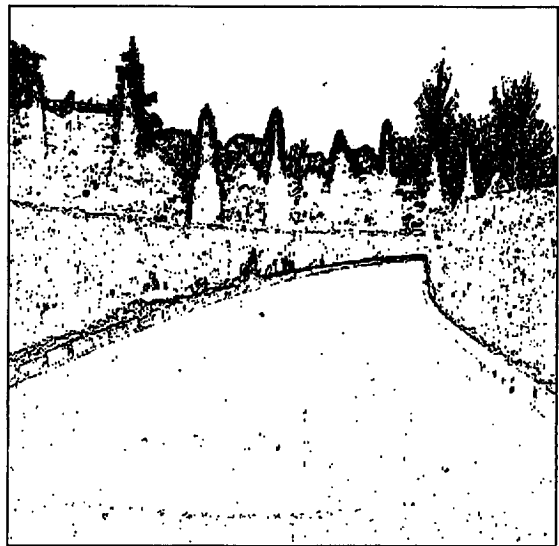
(b)

Fig. 6. Robot motion in environment 1.

efficiently. Finally, images yielded by fuzzy logic inference are binarized.

In most applications, however, there exists much noise in the original image. For example, Fig. 7(a) shows an experimental environment under strong solar reflections. In this case, extraction of white lines becomes difficult. To deal with this problem, we integrate special knowledge on features of white
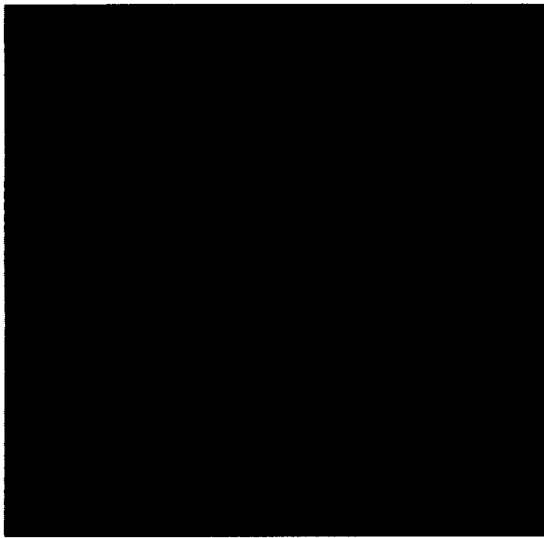
(a)
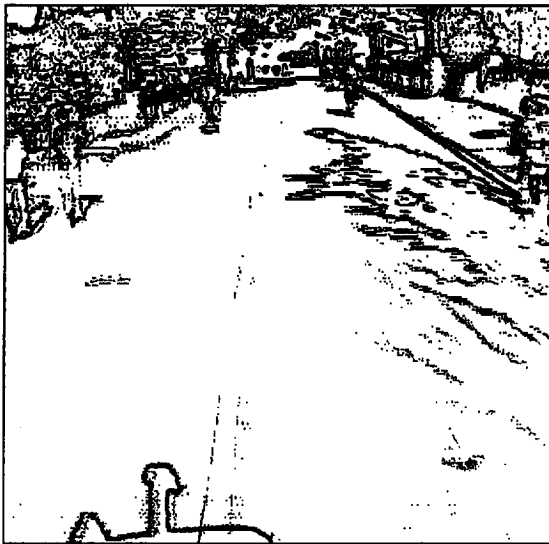
(b)

Fig. 7. Robot motion in environment 2.

line markings into the fuzzy rule base. It is known that a white line marking is bright and has a high contrast. Using this feature, the image is processed in the following way. Each pixel in the image is considered as a candidate white line marking if it has a gray level intensity greater than intensities of its neighboring pixels; and it should belong to a border region. Therefore, we propose the following modi-
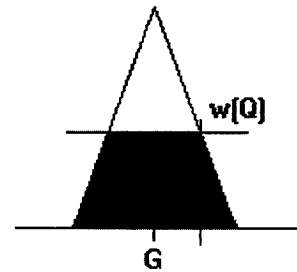

Fig. 8. Membership function of the gray level intensity for white line markings.

fied strategies:

- if a pixel belongs to a border region and it is close to a feature then make it black else make it white
- if a pixel belongs to a uniform region and it is close to a feature then make it white else make it black

In doing it, we define a membership function of the gray level intensity as shown in Fig. 8, and in our experiments, its center point, Q, is chosen as the gray-level intensity 155. Obviously, each pixel, Q, is fuzzified by this membership function, and it contributes a weight $w(Q)$ according to it gray level intensity. In order to extract a white line marking, the weight w_t is modified by the following equation:

$$w_t = \max\{w(i),w(Q)\}. \tag{5}$$
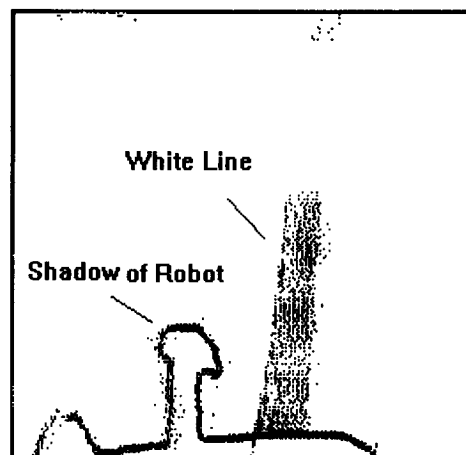

Fig. 9. Recognition of white line markings on the street of Tsinghua University campus.

The following algorithm is proposed to process each pixel, Q:

Step 1: Compute its luminance differences, $Lum\_Dif(i)$ $(i = 1, \ldots, 8)$, with respect to its neighboring pixels;

Step 2: Based on the membership functions defined in Fig. 2, fuzzify the luminance differences, $Lum\_Dif(i)$ $(i = 1, \ldots, 8)$, and compute $w(i)$ $(i = 1, \ldots, 8)$ according to the rule base in Fig. 3;
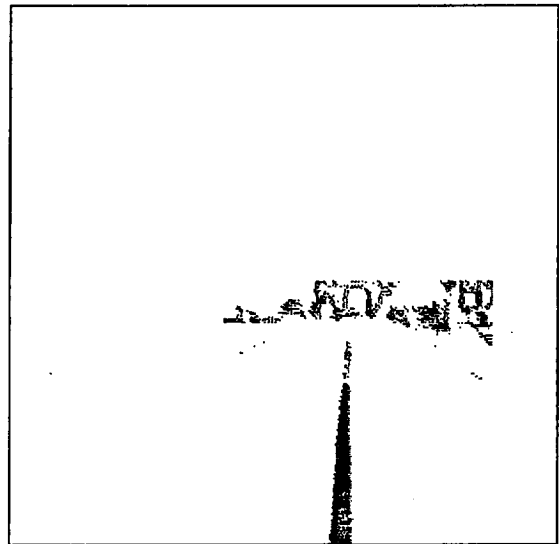
Step 3: Fuzzify this pixel, Q, based on the membership functions of the gray level intensity defined in Fig. 8 to obtain $w(Q)$;

Step 4: Use $w(i)$ and $w(Q)$ to compute w_t using Eq. (5);
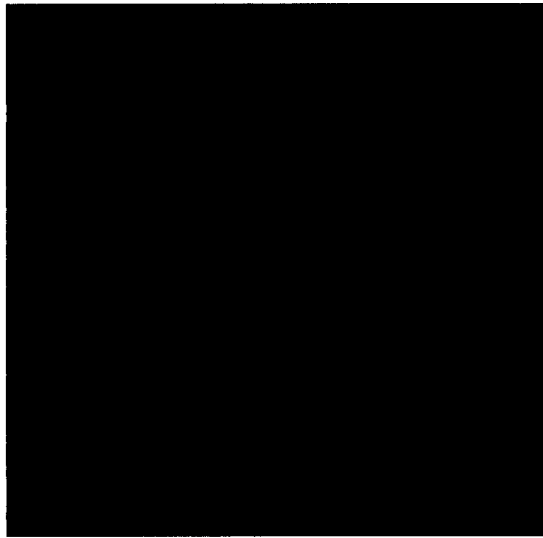
(a)
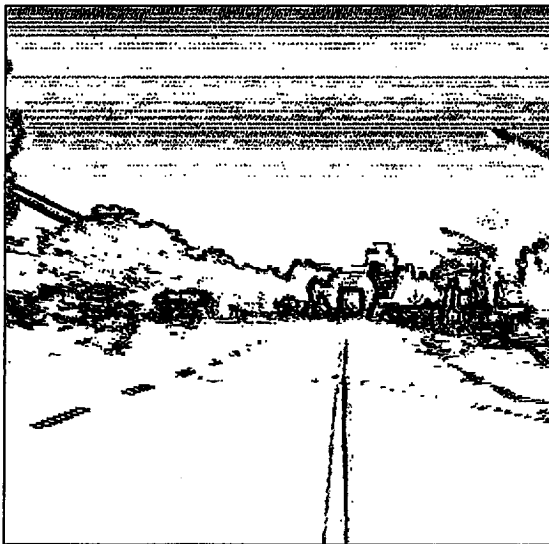
(b)

(c)

Fig. 10. Robot motion in environment 3.

Step 5:   Classify this pixel into white or black using Eq.(4).

5. Experimental results

The experimental environment in Fig. 7 is a street on Thsinghua University Campus. White line mark-

ings are very fuzzy since this street is used for a long time and the image is taken under strong solar reflection. In this case, we integrate the knowledge about white line markings into the fuzzy rule base. In practice, we only process a part of the original image. The result in Fig. 9 shows an extracted while line marking. This white line marking is clear and suitable for robot motion control. For comparison, a
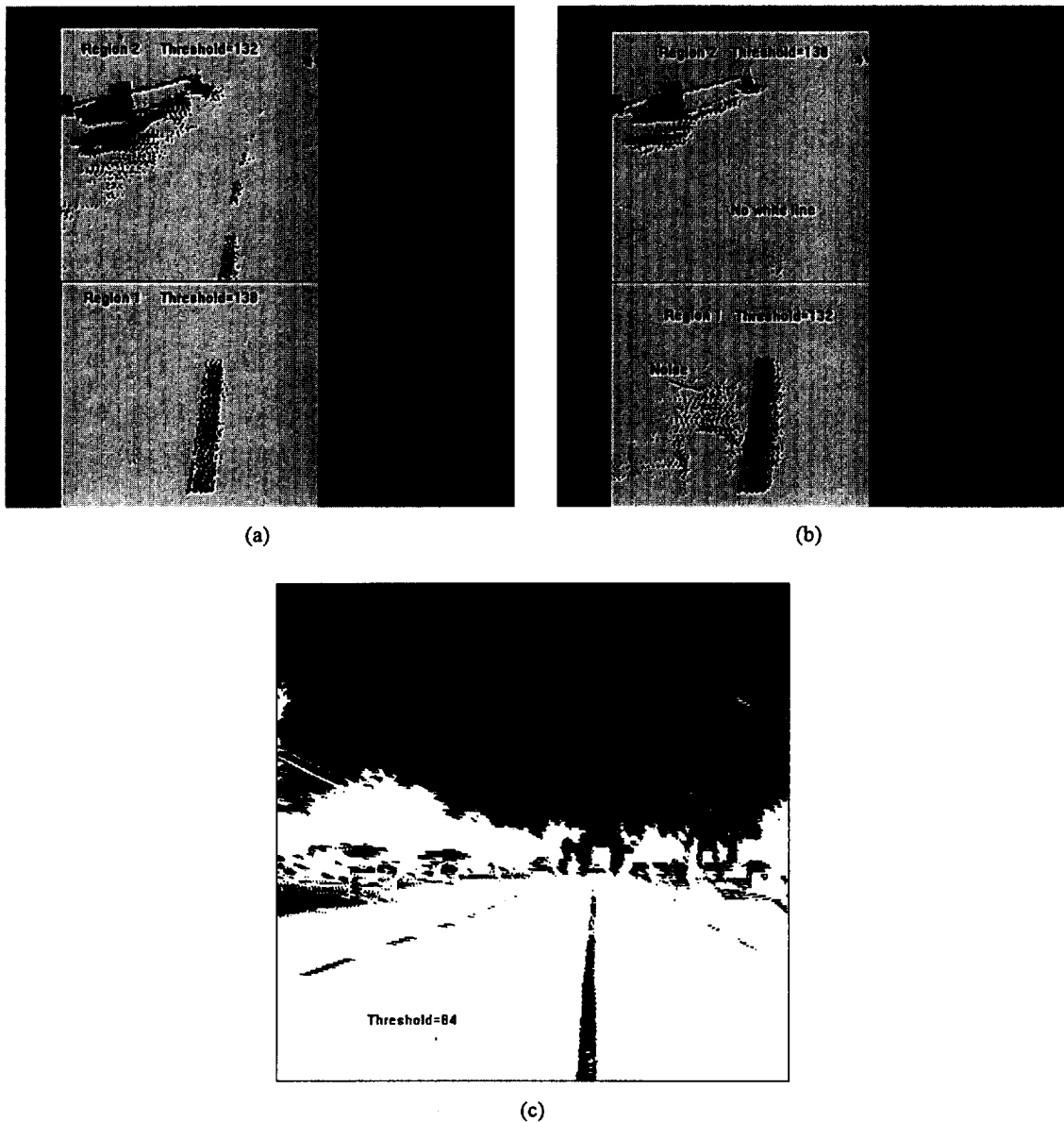

(a)

(b)

(c)

Fig. 11. White line markings detected by the statistical thresholding scheme.

simple statistical thresholding algorithm is implemented for recognition of white line markings. However, we could not get this white line marking for robot motion control due to noise in the original image.

The following set of pictures show another experiment. In this case, we perform our experiments on a normal street outside of Tsinghua University campus, and time of the experiment is close to evening. Fig. 10(a) is the original image taken by the front CCD camera. Fig. 10(b) shows the image processed by an algorithm for general edge detection, and Fig. 10(c) shows the image processed by the proposed algorithm. It can be seen that only the white line markings is the dominant part in Fig. 10(c); whereas other parts that are not important for robot motion are removed from the original image.

6. Discussions and conclusions

In Yao, 1994, a statistical thresholding scheme was tested to detect white line markings. In this scheme, an original image is first filtered, and then a threshold is determined based on the histogram of the image. If the gray level of a pixel is greater than the threshold, this pixel is classified as black; Otherwise, the pixel is classified as white. However, thresholds for edge extraction depend largely on environment changes. Fig. 11(a) and Fig. 11(c) show the images of environments 2 and 3 processed by this scheme. In the images, we give the obtained thresholds. It can be seen that the acceptable thresholds for both of the environments are different. Even in the same image, different thresholds must be determined to process different regions to obtain the desired result. The example in Fig. 11(a) shows that the thresholds for the region 1 (138) and region 2 (132) is different. If we use the threshold 138, which is suitable for region 1, to segment region 2, we can not get white line markings although the change to the threshold 132 is not large. Besides, this scheme is very sensitive to noise. For instance, if we use the threshold 132, which is suitable for region 2, to segment region 1, the final image in Fig. 11(b) shows much noise in it. Although several techniques are used to adaptively compute the thresholds, it is very difficult to determine the optimal thresholds for complex and varying out-door environments. In fact, all the thresholds presented in this paper are obtained by manually modifying the original thresholds computed by the scheme.

The proposed method for the detection of white line markings is not sensitive to noise. It should be noted, that the fuzzy rule base and the membership functions are identical for both the experimental environments in Figs. 7 and 10. Although both the experiments are performed at different places as well as different times during the day, white line markings can be clearly detected. This is very useful for robot navigation in out-door environments. In our implementation, an 512×512 pixel image is first compressed into an 128×128 image, and only a part of the entire image is selected for robot navigation, as shown in Fig. 11(a). The run time of the proposed method for processing such an image is about 150 ms using a 90 Mhz 586 CPU. The computation time of the statistical thresholding scheme is faster than the proposed method, one of the reasons being that in our program a floating point operation is used for fuzzification and defuzzification. The run time can be reduced if the floating point operation is replaced by using an integer operation. In particular, the hardware implementation of the proposed method can be realized by using a special chip to speed up the image processing. In comparison with the statistical thresholding scheme, the proposed method is also easy to integrate with a road-model for extraction of road edges (Li and Jiang, in press).

Acknowledgements

References

Bezdek, J.C., Shirvaikar, M., 1994. Edge detection using the fuzzy control paradigm. In: Proc. 2nd European Congress on Intelligent Techniques and Soft Computing, vol. 1. Verlag der Augustus Buchhandlung, Aachen, Germany, pp. 1–12.

Bezdek, J.C., et al., 1995. Edge detection using fuzzy reasoning and model-based training. In: M. Palaniswami (Ed.), Computational Intelligence – A Dynamic System Perspective, pp. 108–123.

Graef, V., Kuhnert, K.D., 1988. Toward a vision-based robot with a driver's license. In: Proc. IEEE/RSJ Internat. Workshop On Intelligent Robots and Systems, pp. 627–632.

Jain, A.K., 1989. Fundamental of Digital Image Processing. Prentice-Hall, Englewood Cliffs, NJ.

Lepage, R., Poussart, D., 1992. Multi-resolution edge detector, In: Proc. IJCNN.

Li, W., Jiang, X.J., in press. Road recognition for navigation of an autonomous vehicle by fuzzy reasoning, Fuzzy Sets and Systems.

Moura, L., Martin, F., 1991. Edge detection through cooperation and competition. In: Proc. IJCNN, 1991.

Russo, F., Ramponi, G., 1994. Edge extraction by FIRE operators. In: Proc. 3rd IEEE Internat. Conf. on Fuzzy Systems, pp. 249–253.

Tao, C.W., Thompson, W.E., Taur, J.S., 1993. A fuzzy If–Then approach to edge detection. In: Proc. 2rd IEEE Internat. Conf. on Fuzzy Systems, pp. 1356–1361.

Thorpe, C., Herbert, M., Kanade, T., Shafer, S., 1991. Towards autonomous driving. IEEE Exprt 6 (4).

Yao, X.H., 1994. The research of multiple-sensor based local planning and navigation for the autonomous land vehicle. Master Thesis, Dept. of Computer Science and Technology, Tsinghua University.

Zadeh, L.A., 1965. Fuzzy sets. Inform. and Control 8, 338–353.