Proceedings of the 2007 IEEE
Intelligent Transportation Systems Conference
Seattle, WA, USA, Sept. 30 - Oct. 3, 2007

Recognition of Road Contours Based on Extraction of 3D Positions of Delineators

Shoichi Shimizu, Hironobu Fujiyoshi, Hiroshi Sakai and Takeo Kanade

Abstract— **Drivers of vehicles focus their gaze in the direction of movement, try to assess the contours of the road ahead, and try to avoid imminent hazards by steering or braking. However, driving at night is risky since the range illuminated by the headlights and that of visibility is poor. This paper proposes a system that assists the driver by indicating the curves of the road ahead at night. Their curves are estimated from the 3D positions of delineators located on the sides of roads, which are extracted using a circle detection filter. We demonstrated that it was possible to indicate the curves of the road in a simulation experiment.**

I. INTRODUCTION

Intelligent Transport Systems (ITS) have recently been actively researched, and various systems have been proposed throughout the world. The safety driving support is important technology of preventing traffic accidents. Obstacle detection, a lane-keeping method that extracts white road lines, and Adaptive Cruise Control (ACC) have been proposed to assist driving. NAVLAB [1] has involved research on an automatic system of driving, and it is possible to drive automatically using various sensors. Many methods of calculating the outputs from white road lines have been proposed such as estimating the contours of the road ahead [2], [3], controlling driving [4], and estimating the posture (velocity and yaw) of the vehicle [5]. Drivers of automobiles focus their gaze in the direction of movement, try to assess the contours of the road ahead, and try to avoid imminent hazards by steering or braking.

Information detected from white road lines is important to help drivers understand the contours of the road ahead. Methods based on a model, and a clothoid curve or a spline curve have been proposed [6]∼[11] to extract information from white road lines. However, driving at night is risky, since the range of illumination of the headlights and that of visibility is poor.

We propose a system that assists the driver by indicating the curve of the road ahead at night. The contours are estimated from the 3D positions of delineators located on the sides of roads, which are extracted using a circle detection filter. It expects that the headlights are turned to the direction of the curve using our system.

The procedure for our method is described as follows:

S. Shimizu and H. Fujiyoshi are with the Departmet of Computer Science, Chubu University, 1200 Matsumoto-cho, Kasugai-shi, Aichi, 487-8501, Japan { shiyou, hf } @vision.cs.chubu.ac.jp

H. Sakai is with DENSO CORPORATION Corporate R&D Dept.2, 1-1, Showa-cho, Kariya-shi, Aichi, 448-8661, Japan sakai_h@rd.denso.co.jp

T. Kanade is with the Robotics Institute, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh PA 15213, USA tk@ri.cmu.edu

Fig. 1. Examples of delineators.

1) Detection of the delineators
The delineators are extracted using a circle detection filter.
2) Delineator tracking
The delineators are tracked between consecutive frames.
3) Estimates of the 3D positions of the delineators
In the case of using two cameras, the 3D positions of delineators are estimated using stereo vision. In the case of using one camera, the 3D positions of the delineators are estimated from corresponding points, which are obtained by 2) and vehicle travel data (velocity and yaw of vehicle) using stereo vision.
4) Recognition of road contours
The road contours are estimated by fitting a clothoid curve and a cubic function from the 3D positions of the delineators.

II. DELINEATORS

Delineators complement the vision of drivers when they are assessing curves in the road ahead. Their headlights illuminate the road ahead when driving at night. However, the range that can be clearly seen in the headlights is limited, and it is difficult to obtain a baseline for travel. Street-lighting fixtures solve the lighting problem to some extent. However, setting up and maintaining Street-lighting fixtures on the road for more than limited amounts of traffic involves considerable expense. In contrast, delineators placed at the sides of roads are low expense. Therefore, the delineators are particularly useful at night. Figure 1 shows examples of delineators.

III. DETECTION OF THE DELINEATORS

Delineators reflect the light of vehicles at night, and the reflected light is circular. We therefore propose a method that can be used to detect delineators at the sides of roads by extracting circles.

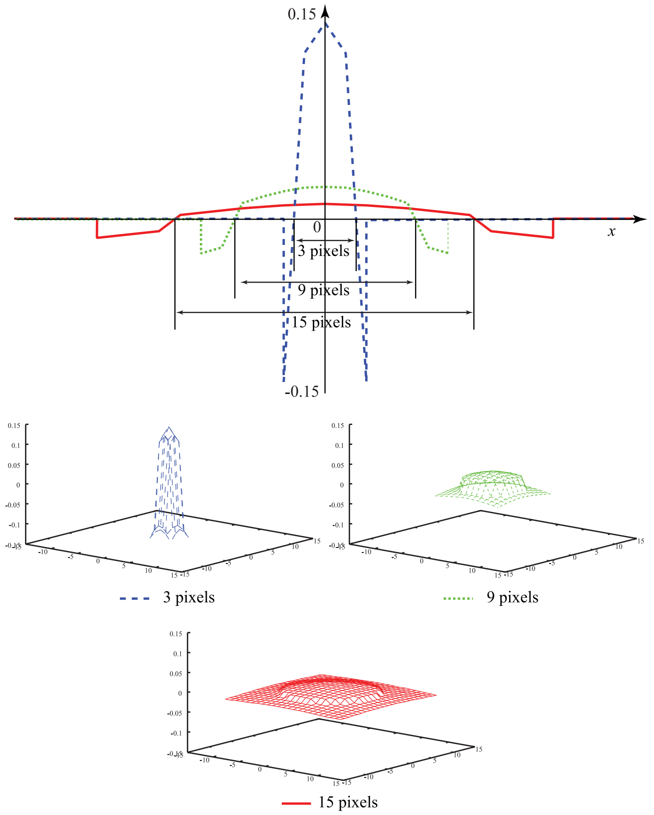


Fig. 2. Examples of filters.

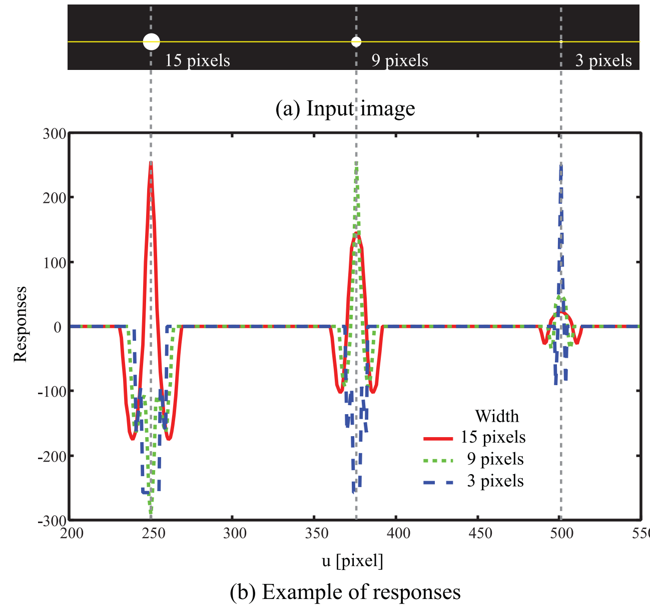


Fig. 3. Examples of responses.

A. Circle detection filter

The circle detection filter was created based on a normal distribution. The normal distribution is defined by

$$G(p,q) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{p^2+q^2}{2\sigma^2}\right). \quad (1)$$

If the radius of the reflected light is $\sigma = r$, the filter is defined by following equation.

$$F(p,q) = G(p,q) - \frac{1}{2\pi\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right). \quad (2)$$

The filter value is normalized to have an average value of zero (the sum of the positive value is 1 and the sum of the negative value is -1). If m_+ is the mean of the positive value and m_- is the mean of the negative value, the filter is given by

$$Filter(p,q) = \begin{cases} \frac{F(p,q)}{m_+} & if \quad F(p,q) > 0 \\ \frac{F(p,q)}{m_-} & otherwise \end{cases}. \quad (3)$$

A high response can be obtained from the center of the circle when the filter convolute a circle with a radius of r. The responses have minus values in areas surrounding the center. The light reflected from different radii can be detected by changing the radius, r, of the filter. Figure 2 shows filters 15, 9, and 3 pixels in diameter. Circles that are 15, 9, and 3 pixels in diameter were created as shown in Fig. 3(a). Figure 3(b) shows the responses on yellow line when the filter convolute a circle. We can see that the response has the highest value when the pixel diameters of the filter equals the pixel diameters of the circle in the image. Therefore, the highest value indicates the center of reflected light.

B. Detection of center of the reflected light

This subsection describes the procedure for calculating the center of reflected light.

1) Apply the 15-pixel-diameter filter to the image.
2) Search responses that are larger than 0.
3) Search the position that has the highest response from the point obtained by 2).
4) Calculate an aspect ratio that is the area of the response, which is larger than a threshold within 15 pixels in diameter. The center of the area is the point obtained by 3).
5) **15-pixel-diameter filter**
   If the aspect ratio is close to one, the point is determined to be the center of the reflected light. Then, the response of the center is kept within 15 pixels in diameter
   Filters 9 and 3 pixels in diameter
   If the aspect ratio is close to one and the response is larger than the maintained value, the point is determined to be the center of reflected light. The response of the center is then maintained within the diameter
6) Return to 2).
7) The next filter is applied after the center has been searched from the entire image. Then, return to 2).
8) The search finishes if all the filters are applied.

The center of reflected light can be detected by these processes.

(a) Response image

(b) Weight map image

Fig. 4. Weight calculation.

IV. Delineator tracking

The reflected light is tracked using mean shift [12]~[14]. The mean-shift algorithm is used to calculate the gradient in the surroundings of the initial position of f(x), and then the center of the area moves toward a higher gradient, and f(x) becomes the maximum position around the initial value. The feature is calculated at the bounding box that clips the object being tracked in mean-shift tracking. The object can be tracked by calculating the amount of movement using mean shift.

A. Weight map calculation

Weight map w_i of surrounding pixel x_i at center x of the previous window is calculated based on response $Res_{num}(num$: number of filter) obtained by the circle detection filter. The weight map is made up of Res_{num}, input image I, and the kernel function. The number of responses is the number of filters that can track the object.

$$w_i = Res_{num}(x_i)I(x_i)Kernel(p,q) \quad (4)$$

$$Kernel(p,q) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{p^2+q^2}{2\sigma^2}\right). \quad (5)$$

B. Distance calculation

The distance, Δx, (mean-shift vector) is calculated by

$$\Delta x = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i} - x = \frac{\sum_{i=1}^{n} w_i(x_i - x)}{\sum_{i=1}^{n} w_i}, \quad (6)$$

where $x_i (i = 1, \cdots, n)$ represents the pixels surrounding the center of window x and w_i is the weight map. The window moves from the weight map distribution surrounding the window.

C. Tracking procedure

Let $\hat{x_0}$ be the initial center of the window. The procedure is described as follows:

Step 1 A weight map $w_i(i = 1, \cdots, n)$, which has center $\hat{x_0}$ of the window, is calculated in the present frame

Step 2 Distance Δx is calculated with Eq. (6) as $\hat{x_1} = \hat{x_0} + \Delta x$

Step 3 If distance Δx is larger than the threshold, Steps 1 and 2 are repeated as $\hat{x_0} \leftarrow \hat{x_1}$

If Δx is lower than the threshold, $\hat{x_1}$ is determined to be the center of the window in the present frame.

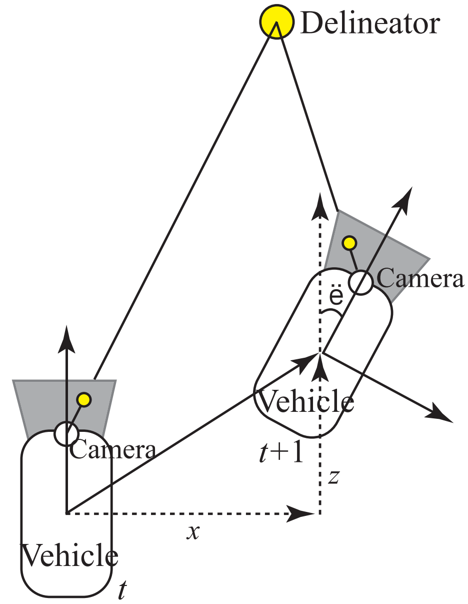


Fig. 5. Stereo vision between consecutive frames using data.

V. Estimation of 3D positions of delineators

The contours of the road ahead are estimated using the 3D position of the delineator. If two camera (stereo camera) are mounted on the car, the 3D position of an object can be calculated using stereo vision. In the case of using one camera, our system calculate the 3D position using information on vehicle-travel data. This data include the movement distance to the x coordinate, the movement distance to the y coordinate and the yaw angle in each frame. The concept of stereo vision between consecutive frames is illustrated in Fig. 5. The relation of the positions of the camera for previous and present frames is assumed to be known. Therefore, the positions of the previous camera and the present one are assumed to be those of two cameras; the 3D position of the delineator is calculated using stereo vision.

VI. Road-contour estimation

A. Delineator model

The method of estimating the contours of the road ahead should take its design into account to recognize it from the delineators. We therefore applied the delineators to a clothoid curve, which is one of the elements of the method of road design, and the contours of the road ahead were recognized from the parameters of this curve.

B. Definition of coordinates

Figure 6 shows how the coordinates in this system are defined. We assumed that the 3D positions would be calculated using stereo vision in this system.

world coordinate The world coordinate has a constant original point (x_w, y_w, h_w).

camera coordinate The camera coordinate changes each frame, the optical axis of the camera is y, and the camera coordinate is defined by (x_c, y_c, h_c). The camera coordinate can transform to the world coordinate using rotation matrix $\boldsymbol{R}$ and translation vector $\boldsymbol{T}$.

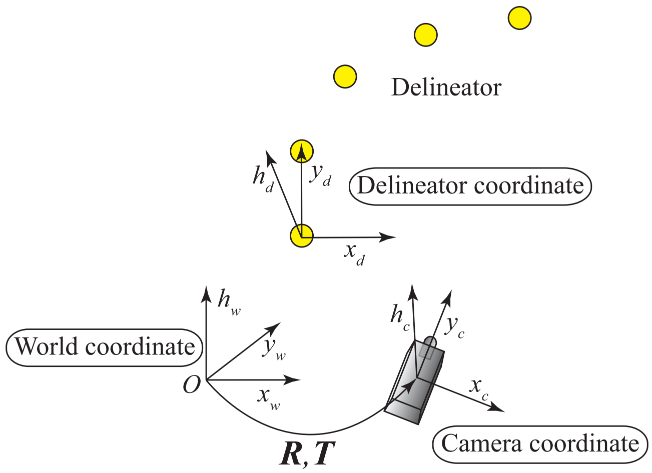


Fig. 6. Definitions of three coordinates.

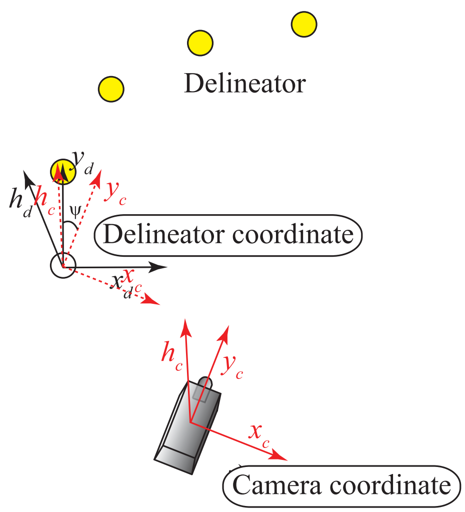


Fig. 7. Transform coordinates.

delineator coordinate The delineator coordinate changes each frame, and the y axis along the delineator. The delineator coordinate is defined by (x_d, y_d, h_d).

C. Definition of the delineator model

The group for the delineator is given by:

$$x = x(s), \quad y = y(s), \quad h = h(s), \tag{7}$$

where s is intervening variable. $x(s)$ and $y(s)$ are defined by the clothoid curve as follows:

$$\begin{cases} x(s) & := x_0 + x_1 \frac{s^2}{2} + x_2 \frac{s^3}{6} \\ y(s) & := s \end{cases}, \tag{8}$$

where x_0 is the clothoid starting point, x_1 is the horizontal curvature, and x_2 represents variations in the horizontal curvature. Moreover, $h(s)$ approximates the cubic function

$$h(s) := h_0 + h_1 s + h_2 \frac{s^2}{2} + h_3 \frac{s^3}{6}. \tag{9}$$

Our goal was to estimate $x(s)$, $y(s)$, and $h(s)$ using the 3D position of a delineator.

D. Estimation of parameters

The 3D position was applied to the clothoid curve and the cubic function, since this system outputs the 3D position for a delineator. The contours of the road ahead can then be recognized from the parameters that are obtained. The method is described as follows:

Step 1 Transform to delineator coordinate
The intervening variable, s, has the relation of $y(s) = s$, which is expressed as the Euclidean distance. If the delineators are aligned on a straight line, the Euclidean distance equals $y_c(i)$. Therefore, $y_c(i)$ can be used for fitting. However, if the delineators are on a curved line, there are errors between the Euclidean distance and $y_c(i)$. $\boldsymbol{T}$ is therefore necessary to transform the delineator coordinate before the delineator is applied to a clothoid curve (Fig. 7). The s axis is approximated from the camera coordinate to the delineator coordinate using the calculated rotation angle, ψ. All 3D positions for the delineator are transformed by:

$$\boldsymbol{X}_d = \boldsymbol{X}_c \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} + \boldsymbol{T}_c, \tag{10}$$

where $\boldsymbol{X}_d = [x_d \; y_d \; h_d]$ is the 3D position on the delineator coordinate, $\boldsymbol{X}_c = [x_c \; y_c \; h_c]$ is the 3D position on the camera coordinate, and ψ and $\boldsymbol{T}_c$ are given by:

$$\psi = \text{atan}(x_c(0)/y_c(0)), \quad \boldsymbol{T}_c = [x_c(0) \; y_c(0) \; h_c(0)].$$

Step 2 Calculation of all parameters
The road contours are estimated by fitting a clothoid curve and a cubic function from the 3D positions of the delineators as shown in Fig. 8.

(a) Calculation of intervening variable s
The intervening variable, s, in Eq. (8) is determined as the Euclidian distance on the delineator coordinate.

$$s_0 = ||\boldsymbol{X}_d(0)|| \tag{11}$$

$$\Delta s_{i,i-1} = ||\boldsymbol{X}_d(i) - \boldsymbol{X}_d(i-1)|| \tag{12}$$

$$s_i = s_{i-1} + \Delta s_{i,i-1} \quad (i = 1, 2, 3 \cdots n)$$

(b) Clothoid curve fitting
A clothoid curve is applied to the 3D position of the delineator on the $x - y$ coordinate using the Levenberg-Marquardt (LM) method. The parameters, x_0, x_1 and x_2, are calculated.

(c) Cubic function fitting
The cubic function is applied to the 3D position (y, h) of the delineator on the $h - y$ coordinate using the LM method. The parameters, h_1, h_2 and h_3, are calculated.

Step 3 Estimation of road contour
The parameters for the clothoid curve are calculated in Step 2. Parameter x_0 is the stating point. Parameter x_1 is the curvature, and this indicates the amount of curve in nearby areas. Moreover, x_2 represents variations in curvature, and this parameter indicates the amount of curve in distant areas. Therefore, the direction of the curve can be detected from these parameters.

Step 4 Transform to camera coordinate

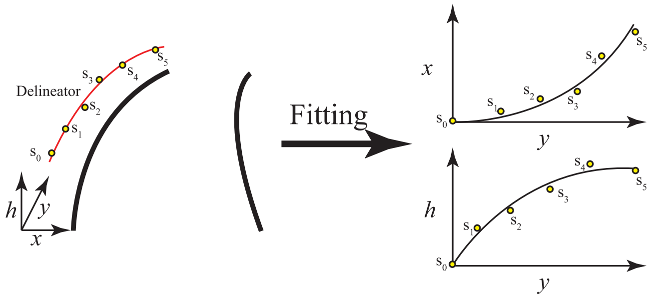


Fig. 8. Clothoid curve and cubic function are applied to 3D position of delineator.

The delineator coordinate is transformed to the camera coordinate, since the present coordinate is the delineator coordinate transformed by Step 1.

$$\boldsymbol{X}_c = \boldsymbol{X}_d \begin{bmatrix} \cos(-\psi) & \sin(-\psi) & 0 \\ -\sin(-\psi) & \cos(-\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} - \boldsymbol{T}_c.$$

Step 5 Transform to world coordinates

Since, the coordinates of parameters (x_0, x_1, x_2) of the clothoid curve and parameters (h_1, h_2, h_3) of the cubic function are the camera coordinate, they are transformed from camera coordinates to world coordinates. Rotation matrix $\boldsymbol{R}$ and translation vector $\boldsymbol{T}$ are needed in this process, and these parameters are calculated from ego-motion or gyro.

$$\boldsymbol{X}_w = \boldsymbol{R}\boldsymbol{X}_c^T + \boldsymbol{T}, \qquad (13)$$

where $\boldsymbol{X}_w = [x_w \ y_w \ h_w]$ is the 3D position on the world coordinate,

VII. Experiments

A. Experiment on detection of delineators

We conducted an experiment in which light reflected from a delineator was detected from the image that was obtained from the camera. In this sequence, the number of the delineators are 2,298, and the road is constructed with the right and left curves.

We used three kinds of filters, i.e., 15, 9, and 3 pixels in diameter. Figure 9 shows an example of delineators being detected. The red circle is the filter of 15 pixels, the green circle is the filter of 9 pixels and the yellow circle is the filter of 3 pixels. You can see that the centers of the delineators have been extracted. Table I lists the percentages for detected correct, false positives, and false negatives for the delineators. The number of the detection of the correct answer is defined as C, the number of the false positive is defined as fp and the number of the false negative is defined as fn. These rate are calculated by:

$$correct\ rate = \frac{C}{C+fn} \times 100 \qquad (14)$$

$$false\ positives\ rate = \frac{fp}{C+fp} \times 100 \qquad (15)$$

$$false\ negative\ rate = 100 - correct\ rate. \qquad (16)$$

We found that 92.2 % of the delineators were correctly detected, which is quite high. From the Table I false negative

Fig. 9. Example of detection of delineators.

TABLE I

RESULTS FOR DETECTION DELINEATORS [%]

Correct	False positive	False negative
92.2	21.4	7.8

is 21.4 %. The cause of the false detection is occurred a faraway center line (dashed line) and a pole. In the case of the faraway dashed line, the line is extracted, since appearance is very similar to the delineators. It is expect to be able to reduce the false detection, since the appearance of the pole is different clearly.

B. Experiment on delineator tracking

We conducted an experiment on delineator tracking between consecutive frames. The input image was the same image as in the previous experiment. Figure 10 shows an example of tracking. We found that 91.9 % of the delineators were correctly tracked. It is clear that tracking is very efficient from these results.

C. Experiment on 3D position of delineators estimation

To evaluate 3D position, we created J-curve with eight kinds of curvature (15, 30, 60, 100, 150, 280, 460, 710 m). Then, the delineators were placed based on road rules. The 3D positions of the delineator were estimated using stereo vision, and clothoid curve was applied to the 3D positions of the delineator. Figure 11 shows an example of 3D position estimation. We can see that the clothoid-curve fitted the 3D position, and it is possible to recognize the curve using the parameters of clothoid curve.

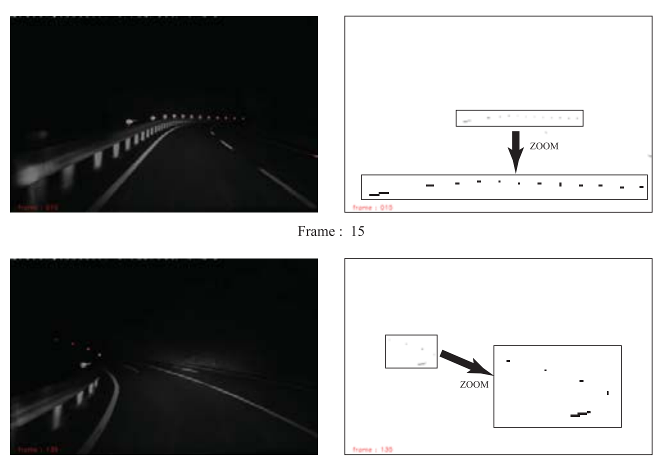
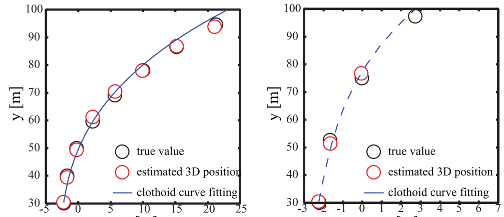
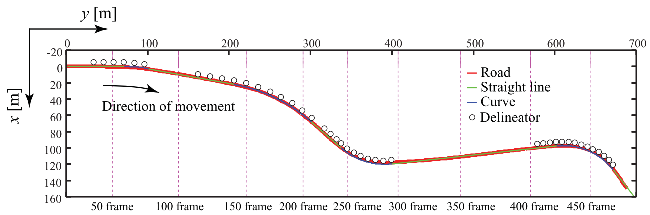


Fig. 10. Example of delineator tracking.

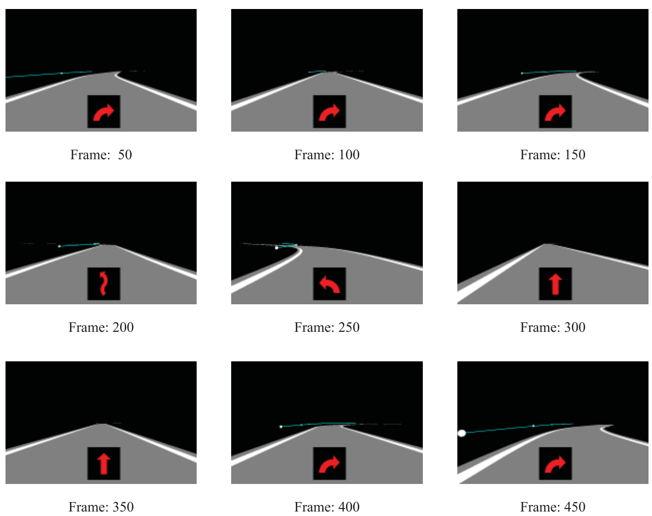


(a) curvature: 100 m (b) curvature: 460 m

Fig. 11. Example of 3D position estimation.



(a) Road created using vehicle travel data



(b) Example of road recognition

Fig. 12. Road created and example of road recognition.

D. Experiment on road recognition

We evaluated the method we propose by simulation. The road in the simulation space was created from vehicle travel data from odometer, and the delineators were placed based on road rules. Figure 12(a) shows the road created in the simulation space.

We assumed that the 3D position of the delineators had already been detected. Therefore, the contour of the road was only recognized by using clothoid-curve fitting.

Figure 12(b) shows examples of road recognition from the parameters of the clothoid curve. The arrow in Fig. 12(b) indicates the curve in the road ahead. The accuracy rate can be calculated using ground truth, since the curve was created in the simulation space. The accuracy rate is 85.6 %. When the curve was S-shaped curve, the recognition was often failed. The curve was distinguished using the thresholds of the parameters. Therefore, the recognition rate depend on the thresholds.

VIII. CONCLUSIONS AND FUTURE WORKS

A. Conclusion

We proposed a system that assists drivers by indicating the contours of the road ahead at night. The contours are estimated from the 3D positions of delineators placed at the roadside and are extracted using a circle detection filter. We demonstrated that it was possible to indicate the curves of the road in a simulation experiment. Therefore, this system should be useful for enabling drivers to predict the contours of the road ahead.

B. Future work

It is possible to extract the light reflected from delineators beyond about 100 m. However, it is difficult to calculate the precise 3D positions of delineators beyond about 100 m. The accuracy of calculating 3D positions is expected to be improved by taking the structure of delineators into consideration.

The processing time of our method is slow, since our method perform the convolution three times. It expects that speed up the convolution speed by parallel processing.

REFERENCES

[1] C. Thorpe, M. Hebert, T. Kanade, and S. Shafer: “Vision and navigation for the Carnegie-Mellon Navlab”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 10, No. 3, May, 1988, pp. 362 - 373.

[2] R. Aufrère, R. Chapuis and F. Chause: “A model-driven approach for real-time road recognition”, Machine Vision and Applications (2001) 13: pp. 95-107, 2001.

[3] Nedevschi, R..Schmidt, T. Graf, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol: “3D Lane Detection System Based on Stereovision”, IEEE Intelligent Transportation Systems Conference, pp. 161-166. 2004.

[4] D. Pomerleau: “RALPH: Rapidly Adapting Lateral Position Handler”, Proc. of 1995 Symposium on Intelligent Vehicles, pp. 506-511, 1995.

[5] Jun Adachi and Jun Sato: “Estimation of Vehicle Position from Uncalibrated Cameras”, The Institute of Electronics, Information and Communication Engineers, Vol. J89-D, No. 1, pp. 74-83, 2006.

[6] C.J. Taylor, Jitendra Malik and Joseph Weber: “A Real-Time Approach to Stereopsis and Lane-Finding”, Intelligent Vehicles 1996, pp 207-212.

[7] A. Takahashi and Y. Ninomiya: “Model-Based Lane Recognition”, Proceeding of the 1996 IEEE intelligent vehicles symposium, pp. 201-206, 1996.

[8] 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, pp. 62-81, 1998.

[9] D. Khosla: “Accurate estimation of forward path geometry using two-clothoid road model”, Intelligent Vehicle Symposium, 2002. IEEE Vol. 1, Issue, 17-21 pp. 154 - 159, 2002.

[10] Y. Wang, D. Shen and E. K. Teoh: “Lane Detection Using Catmull-Rom Spline”, Proc. of intellignet Vehicles Symposium, Vol. 1, pp. 51-57, 1998.

[11] J. Goldbeck and B. Huertgen: “Lane Detection and Tracking by Video Sensors”, Proc. of IEEE International Conference on Intelligent Transportation Systems, pp. 74-79, 1999.

[12] R. Collins: “Mean-shift blob tracking through scale space”. IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 234-240, 2003.

[13] D. Comaniciu and P. Meer: “Mean shift analysis and applications”. IEEE International Conference on Computer Vision, Vol. 2, pp. 1197-1203, 1999.

[14] D. Comaniciu, V. Ramesh, and P. Meer: “Real-time tracking of non-rigid objects using mean shift”. IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2. pp. 142-149, 2000.