Procc. of 10th International Conference on Information Technology, December 17-20, 2007, Rourkela, pp. 38-40

A Fast Edge Detection Algorithm for Road Boundary Extraction Under Non-uniform Light Condition

Aurobinda Routray
Deptt. of Electrical Engineering,
I.I.T. Kharagpur-721302, India

Kanungo Barada Mohanty
Deptt. of Electrical Engineering,
N.I.T., Rourkela-769008, India
kbmohanty@nitrkl.ac.in,
barada5@rediffmail.com

Abstract

In this paper a fast edge detection algorithm based on a simple logic has been implemented for road boundary detection in non-uniform light condition. Road images taken in the campus have been used to test the algorithm. First the image samples are segregated into different segments depending on the intensity. Subsequently standard edge detectors are applied to extract the edges in each of the segments. A logical operation between the edges of the different segments brings out the edges of the final image.

kbmohanty@nitrkl.ac.in,
kbmohanty@nitrkl.ac.in,
barada5@rediffmail.com

1. Introduction

Road boundary detection is one of the guiding factors in autonomous and semi-autonomous vehicles. Simple edge detection algorithms that extract all the edges found in the area of interest as well as in the surroundings, are not quite efficient under non-uniform light and noisy measurement conditions, something unavoidable for on-board cameras.

A single camera with Conic Projection Sensor System has been used in [1], while a two camera system has been used in [2] to get real-time images and the 3-D perspective is removed for vehicle guidance. In [3], Hough Transform has been used to extract 3-D road boundaries in the images coming from a stereo-vision system. Geometrical features are used in [4] to find road boundaries out of a set of many image segments. A multi order line segment description of the road boundary has been achieved through a statistical test, in [5]. The edge location and edge orientation are also considered in [5]. In [6], low-level structures are extracted via a non-linear transform and it proposes an algorithm for segmentation of images at multiple scales.

A simple and fast method based on segmentation followed by edge detection and logical operation has been suggested in this paper for road-edge detection. Only the lower half of the image needs to be used to predict the immediate future coordinates for guidance. In addition to that, this algorithm is insensitive to the unevenness and roughness in the roads, which manifest as regions of non-uniform intensity.

2. The Algorithm

The block diagram representation of the algorithm is shown in Fig. 1. The actual image-data is processed through two independent steps and finally the combination of these two steps gives the output. Different stages of this algorithm are briefly explained

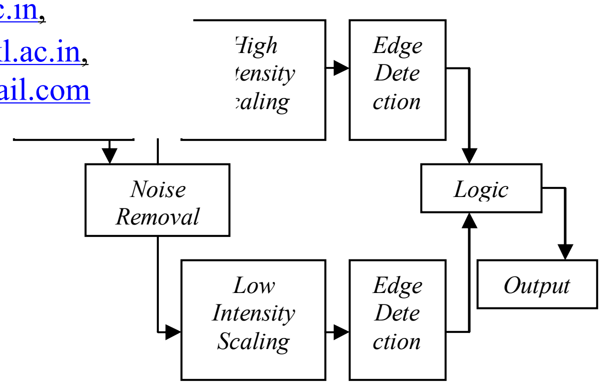

Fig.1 Block diagram representation of the algorithm

2.1 Image pre-processing

Let the image be represented as, $I(x, y)$, with x and y being the two coordinate directions. Median filtering, proposed in [7], is used for noise elimination. Fig. 2(a) is the original image of road while Fig. 2(b) is the median filtered image. Median filter can be represented as,

$$v(m,n) = \text{median}\{ I(m-k, n-l), (k,l) \in W \}$$

Where, W is a suitably chosen window.

The intensity scaling is implemented to eliminate the effect of variation in light as well as, the influence

DOI 10.1109/ICIT.2007.9

of the background. Scaling is done taking the selected value of upper and lower threshold as the reference. The value of upper and lower threshold varies between '0' and '255', respectively the smallest and largest gray scale intensity values of any image pixel. Scaling has the form,

$$i(x,y)_{a,b} = \begin{cases} 1.\ I_{min} & \text{if } i(x,y) <= a \\ 2.\ i(x,y) \times I_{max} / (b-a) & \text{if } a < i(x,y) < b \\ 3.\ I_{max} & \text{if } i(x,y) >= b \end{cases}$$

Where, I_{max} = maximum possible value of intensity
I_{min} = minimum possible value of intensity
$i(x, y)$ = actual value of intensity at a point (x, y)
a = lower threshold for intensity scaling
b = upper threshold for intensity scaling
Also, $0 <= I_{min} < a < b <= I_{max}$

Fig. 2(a) The original image of road

Fig. 2 (b) The median filtered image of (a)

2.2 Edge detection

Prewitt filter [8, 9] and Sobel filter [9], are used for boundary extraction. Performances of both the filters are almost similar. The output image after boundary extraction can be represented as:

$$E(x, y) = v(x, y) \times h(x, y)$$

$E(x, y)$ is the image data containing extracted edge, $v(x, y)$ is the original image data, $h(x, y)$ is the filter matrix.

In this algorithm, all the above mentioned steps are implemented on the image data in two independent ways which differ only in the value of upper and lower intensity thresholds. While in one process low values of intensity are scaled out, the same happens with high intensity values in the other process.

Scaling out low intensity values leaves out a large portion of the road black in the image data so produced. Hence, the edges due to disturbances are minimized in the region within the road boundaries. The same happens with the region outside the road boundary when we scale out high intensity values, only difference being that this time the region with intensity greater than the upper threshold becomes white. This happens due to the road being darker than the surroundings (this assumption is made because this is the most likely condition to be faced by the vehicle during its course of motion). This method also takes care of the case where the road is lighter in intensity than the surrounding because of the opposite effect.

Thereafter, logical AND operation of the two output image data gives us the required edge as the region near the road boundaries is most likely the only common region in the two images and hence the edges in other regions get discarded as AND operation can take only simultaneously high pixel values in both the images.

3. Implementation

In order to test the performance, the algorithm has been implemented on Matlab as well as on the C++ platform. Implementing the algorithm with varying thresholds, which depend on the gray scale intensity level, we get the output processed images having clear road boundaries.

At first, the algorithm is implemented on Matlab and results are shown in Fig.3 and Fig.4. Fig. 3 shows the processed image of the image in Fig. 2 (a). Fig. 4-(a) shows an image different from Fig. 2(a) taken from a different angle. Fig. 4-(b) is the processed images showing road boundaries. As can be seen in the processed image, the effect of the edges present in the surroundings has been minimized to a large extent.

Fig. 3 Boundary extracted image of the original image in Fig.2 (a), implemented in Matlab

Next, this algorithm is implemented on the C++ and the result is shown in Fig. 5, which is the extracted image of the road image in Fig.4(a). This implementation shows complete elimination of top half of the image in the output. This is due to similarity in the grey level of that region and with suitable value of upper and lower intensity scaling thresholds this has been eliminated. The value of upper threshold has been taken to be 110 and 45 as lower threshold.

Fig. 4 (a) Image of road from a different angle

Fig. 4 (b) Boundary extracted image of 4-(a), implemented in Matlab

Fig. 5 Boundary extracted image of the image in Fig. 4-(a), implemented in C++

4. Results and Discussions

Since, very simple logical operations and linear filters are used, this algorithm is significantly fast. As the noise is mostly impulse noise, the use of median filter fulfills the requirement. Adaptive median filter may do well in cases having the need for very high accuracy. But the size of the filter window has to be right. Larger size than the required filter size may result in smoothing out of the images, resulting in very unclear edges. Smaller sized window may result in insufficient noise removal.

This method is based on the difference in intensity level of the light reflected by the road and the surroundings, which will be present almost always. This makes it more reliable than the non-linear methods predicting the line of motion of road boundary. As it is clear, this algorithm works very well, when light is reflected more by the road. In the extreme case, when the intensity of light reflected by the road is less than that by surroundings, then there is no difficulty with this method, as the upper and lower thresholds for intensity interchange their effect and the road boundary is detected with equal accuracy.

5. Conclusions

This method works well with all types of conditions and has shown its insensitivity to small disturbances, as the areas of unevenness on the road are minimized in the final output image. The images used here are not normal clear images. In spite of that the extraction of edges shows the reliability and accuracy of this method. This method has very high potential for being implemented in future autonomous vehicular motion systems and driver assistance systems, due to its good speed and accuracy. Also, with the development of new architecture day by day that support more and more parallel computation, this method can be seen as the real solution of artificial vision problem.

References

[1] Z. Zhu, S. Yang, G. Xu, X. Lin, D. Shi, "Fast Road Classification and Orientation Estimation Using Omni-View Images and Neural Networks," *IEEE Trans. on Image Processing*, vol. 7, no. 8, pp 1182-1197, Aug. 1998.

[2] M. Bertozzi, and A. Broggi, "GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection," *IEEE Trans. on Image Processing*, vol. 7, no. 1, pp. 62-81, Jan. 1998.

[3] U. K. Sharma, and L. S. Davis, "Road boundary detection in range imagery for an autonomous robot," *IEEE Journal of Robotics and Automation*, vol. 4, no. 5, pp. 515-523, Oct. 1988.

[4] J. P. Tarel, and F. Guichard, "Combined dynamic tracking and recognition of curves with application to road detection," *Procc. of IEEE Int. Conf. on Image Processing*, Sept. 2000, vol. 1, pp. 216-219.

[5] G. Tang, X. Liu, X. Liu, H. Yang, "A road boundary detection method for autonomous land vehicle," *Procc. of the 4th World Congress on Intelligent Control and Automation*, June 2002, Shanghai, vol. 4, pp. 2949-2951.

[6] M. Tabb, and N. Ahuja, "Multiscale Image Segmentation by Integrated Edge and Region Detection," *IEEE Trans. on Image Processing*, vol. 6, no. 5, pp. 642-655, May 1997.

[7] H. Hwang, R. A. Haddad, "Adaptive Median Filters: New Algorithms and Results," *IEEE Trans. on Image Processing*, vol. 4, no. 4, pp. 499-502, April 1995.

[8] J. Scharcanski, and A. N. Venetsanopoulos, "Edge Detection of Color Images Using Directional Operators," *IEEE Trans on Circuits and Systems for Video Technology,* vol. 7, no. 2, pp. 397-401, April 1997.

[9] R. C. Gonzalez, and R. E. Woods, *Digital Image Processing,* Prentice Hall, 2nd Ed. 2002.