Realize a Mobile Lane Detection System based on Pocket PC Portable Devices

Pi-Chih Hsueh, Chun-Wei Yeh, Chao-Han Cheng, Pei-Yung Hsiao[1],
Ming-Jer Jeng and Liann-Be Chang
Department of Electronic Engineering, [1]Department of Electrical Engineering
Chang-Gung University, [1]National University of Kaohsiung
259, Wen-Hwa 1 Rd., Kuei-Shan, Tao-Yuan, Taiwan
Republic of China
http://www.cgu.edu.tw

Abstract: - In this paper, we present a mobile lane detection system (LDS) based on the Pocket PC portable system. The Pocket PC application software with generic 2-D Gaussian smoothing filters which are including the power-of-two approximation algorithm for the Gaussian coefficients is easily to be hardware design. In the lane detection algorithm stage of an image processing flow, the global edge detection is able to transfer the gray level image to binary pattern and shows the edge of the object. Then, using this binary pattern find out the traffic lane location with following algorithm like the peak-finding and grouping, edge connecting, lane segment combination, lane boundaries selection. At last, the lane departure warning algorithm detects the vehicle whether in traffic lane and judges whether sends out the warning. Experimental results both operated under different circumstances and real image sequences will also be presented.

Key-Words: - Mobile lane detection system, Pocket PC portable devices, Generic 2-D smoothing filters, Lane detection algorithm, Global edge detector, Lane departure warning algorithm.

1 Introduction

Traffic accidents have become one of the most serious problems in today's world. Most accidents occur due to drivers' negligence. Many researchers and vehicle makers are developing a driving assistance system and improving the driving safety [1-14]. There are many research topics on driver assistance systems mainly focusing on the lane detection technology since it is basic and has many applications, such as autonomous vehicles [1,4] and the lane departure warning systems[5,8-14]. An important step in developing a lane departure warning system is to have a robust lane detector. The lane detection technique is one of the most important parts in the lane departure warning system. To realize the lane detection, the noise disturbance can greatly affect lane distinction. Noise signals may be easily introduced into the original image during the image acquisition stage, the image compression stage or even the image transmission stage [14]. To eliminate such noise signal and ensure the performance of image, an additional step of noise reduction is required. The noise reduction techniques usually involve averaging the value of pixels reside in a local area and generating a blurred or smoothed image. Numerous software-based and hardware-based edge detection algorithms have been proposed. Most algorithms are able to produce acceptable edge results, but fail as the noise signal increases. Gaussian filter is one of the specialized weighted averaging filters. It has been widely adopted in the field of image processing and computer vision for years, and is known for its image smoothing and noise reduction capability. The Gaussian filter plays an important role in digital image processing tasks such as image segmentation, image blurring, and edge detection. Such filter is usually adopted in the filed of edge detection.

In this paper, we present a mobile lane detection system based on Pocket PC portable device. The algorithm we used is divided into two part, the first part is image preprocessing and the second part is lane detection. To enhance portability and flexibility of the Lane departure warning systems, we will develop a portable real-time lane departure warning system should be cheap, easy to install, and able to be combined with other vehicular consumer devices, such as digital cameras, cell phones, and PDAs. Further, we are going to discuss our algorithm and related research studies

2 Generic 2-D Gaussian smoothing filter and lane detection algorithm

Front level processing of the Lane Detection system need to use the lane detection algorithm. For lane detection algorithm applications, image processing tasks are applied to the real-world images. Noise signals may be easily introduced into the original image during the image acquisition stage, image compression stage or even the image transmission stage. To eliminate such noise signals, and to ensure the performance of the overall image processing flow, an additional step of noise reduction is necessarily required. So we present the Generic 2-D Gaussian Smoothing Filter to reduce noise. The Gaussian filter is one of the specialized weighted averaging filters. It has been widely adopted in the field of image processing and computer vision for years, and is known for its image smoothing and noise reduction capability.

2.1 Generic 2-D Gaussian Smoothing Filter

The values derived from the 2-D Gaussian distribution are mostly complex floating point values, which are difficult to implement in hardware. It would perform slowly even if that certain hardware is implemented. The following shows how we derive the original Gaussian smoothing mask. Defining the index (x, y) of the center element of a 5×5 mask to be (0, 0), and the whole mask to span from (-2, -2) to (2, 2). The value of each element in the 5x5 Gaussian smoothing mask is defined by Gaussian function.

$$G_\sigma(x,y) = e^{-(x^2+y^2)/2\sigma^2} \quad (1)$$

Where x, y = {-2, -1, 0, 1, 2} and σ=[0.1, 5.0]. The summation of the whole mask should be equal to 1 by the definition of Gaussian distribution. Therefore, we need to sum the whole 25 elements and normalize them.

$$S_\sigma = \sum_{x=-2}^{2}\sum_{y=-2}^{2} G_\sigma(x,y) \quad (2)$$

$$G_{n\sigma}(x,y) = G_\sigma(x,y)/S_\sigma \quad (3)$$

To derive the digital-approximated 2-D Gaussian mask for hardware implementation, we use the combination of power-of-two terms to approximate each element, such that it can be implemented by simply shift operations in hardware. Then the intuitive power-of-two approximation algorithm of each of the 25 elements may be written as shown in Fig.1(a) after each element is determined, we still need to make sure that the sum would still be close to 1. Let the value of each approximated coefficient be denoted as Aσ(x , y), and again the sum of twenty five current approximated coefficients would be

$$S_{A\sigma} = \sum_{x=-2}^{2}\sum_{y=-2}^{2} A_\sigma(x,y) \quad (4)$$

The power-of-two approximation for $1/S_{A\sigma}$ is also determined using the algorithm shown in Fig.1(a), except for that the range of i is extended from 7 to 15. An example of our digital-approximated Gaussian mask is shown in Fig.1(b).

```
Approx(val)
  term ← 0
  rtterm[term] ← {0}
  for i ← 0 to 7
      if term < λ
          then if val >= 2^(-i)
                then rtterm[term] ← i
                     val ← val - 2^(-i)
                     term ← term + 1
  return rtterm
```

(a)

$(2^{-3}+2^{-6}) \times$

0	2^{-3}	$2^{-3}+2^{-4}$	2^{-3}	0
2^{-3}	$2^{-2}+2^{-3}$	$2^{-1}+2^{-3}$	$2^{-2}+2^{-3}$	2^{-3}
$2^{-3}+2^{-4}$	$2^{-1}+2^{-3}$	$2^{0}+2^{-4}$	$2^{-1}+2^{-3}$	$2^{-3}+2^{-4}$
2^{-3}	$2^{-2}+2^{-3}$	$2^{-1}+2^{-3}$	$2^{-2}+2^{-3}$	2^{-3}
0	2^{-3}	$2^{-3}+2^{-4}$	2^{-3}	0

(b)

Fig.1 Power-of-two approximation (a) Algorithm and (b) Example of the Gaussian mask for σ= 1.1 and λ= 2.

To describe the errors between the original Gaussian mask and the 2^n-approximated Gaussian mask in a formal way, we define each of the approximated normalized elements in the 2^n-approximated Gaussian mask as:

$$A_{n\sigma}(x,y) = A_\sigma(x,y)/S_{A\sigma} \quad (5)$$

Consequently, the absolute accumulated error is defined as:

$$E_{\sigma\lambda} = \sum_{x=-2}^{2}\sum_{y=-2}^{2}\left|G_{n\sigma}(x,y) - A_{n\sigma\lambda}(x,y)\right| \quad (6)$$

The decision for the acceptable accumulated error affects the algorithm. We adopt to approximate the Gaussian mask. Based on our empirical experiences and cost consideration, we used the two-term's 2^n-approximation 2-D Gaussian mask as the proposed filter.

2.2 Global Edge Detection

The Global threshold edge detector uses a 3×3 mask to detect the difference between each pixel. It needs to obtain mean value and variance of night pixels to decide the edge points. In order to implement by logic circuit, we modified it as equation (8)-(9), which has a very little difference between the original value and the modified one.

$$\mu_{x,y} = \left(\frac{1}{8} - \frac{1}{64}\right)\sum_{\Delta x=-1,\Delta y=-1}^{\Delta x=1,\Delta y=1} g(x+\Delta x, y+\Delta y) \quad (7)$$

Then we use the average difference of a 3×3 mask to replace the variance value of the original algorithm.

$$\delta_{x,y} = \left(\frac{1}{8} - \frac{1}{64}\right)\sum_{i=1}^{9}\left|x_i - \mu_{x,y}\right| \quad (8)$$

Finally, we use the following procedure to make pixels' value in a 3x3 mask become binary.

$$g(x+\Delta x, y+\Delta y) = \begin{cases} 1, \delta_{x,y} \geq Tn \\ 0, \delta_{x,y} < Tn \end{cases} \quad (9)$$

After we modified the original algorithm, we can easily implement it by logic circuit. Furthermore, although Global threshold edge detector is a threshold value based algorithm, the configuration of Global threshold value, Tn, is much easier than that of peak finding based algorithm due to the insensitive characteristic.

2.3 Peak finding and grouping

The processing frame will become a binary image after edge detection procedure, then, the original peak finding algorithm need to be modified. As shown in Fig. 2, we have to define three variables: Ps, Pe, Pp, where Star point(Ps) is the position of first climbing up point, End point (Pe) is the position of last climbing down point, and Peak point (Pp) is the middle point between Ps and Pe. After the edge detection procedure, two sides of lane mark may be detected. Therefore, we will combine two close peak point to a new point as shown in Fig.2. After finding peak points in the image, we then group the peak points, which may locate at the same lane boundaries. We use a 9×11mask to group peak, which is shown in Fig.3. We make every peak point to be the central point of a 9×11 mask, and let other points in the mask to be the same group like as the central point, and then repeat the same operation again and again.

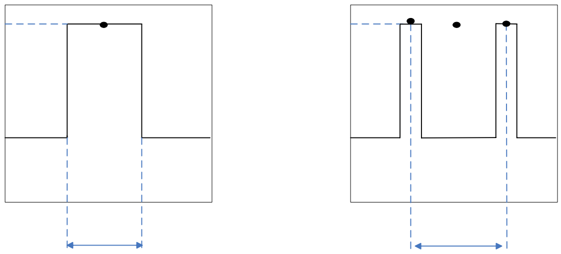

Fig.2 Peak point and combination of two near peak points.

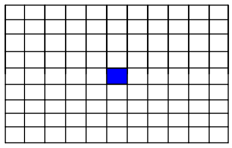

Fig.3 The grouping mask 9×11.

2.4 Edge Connection

The selected peak points are collected and separated as many as line segment by employing the least square method. Each produced line segment is defined as below:

$$L(P_L(X_L, Y_L), P_U(X_U, Y_U), b_0, b_1) \quad (10)$$

Where P_L is the top point of line-segment, P_U is the bottom point of line-segment, b_0 is the intercept of line-segment, and b_1 is the slope of line-segment. The b0 and b1 are generated by equation (11)-(12):

$$b_0 = \overline{X} - b_1\overline{Y} \quad (11)$$

$$b_1 = \frac{n\sum_{i=1}^{n} x_i \cdot y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n\sum_{i=1}^{n} y_i^2 - \left(\sum_{i=1}^{n} y_i\right)^2} \quad (12)$$

3. Architecture of lane departure warning system

In this section we exemplify each state of the lane detection algorithm result and present the software architecture on our platform. The Lane detection application architecture shows in Fig.4. The lower level of the software architecture is portable hardware. Above handhold hardware level is board support pocket like bootloader, device drivers, and OEM Adaptation Layer. The intermediate level is the kernel, device manager, GEWS, and service manager of operation system. Then, the middle level between the application API and lane departure warning application is a virtual machine level.

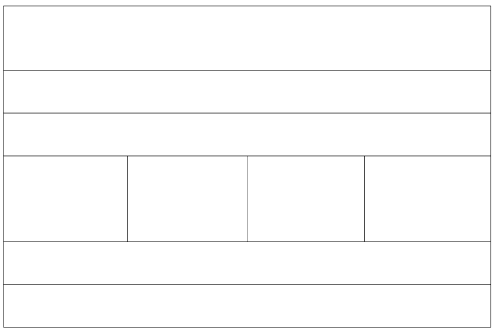

Fig.4 The architecture of lane detection application software built on WinCE mobile OS.

Fig.5 shows the traffic lane image processing flow chart. The input image is processed according to the following order: generic 2-D Gaussian smoothing filter, the global edge detection, the peak-finding and grouping, the edge connecting, line segment combination, lane boundaries selection, lane detection and display results.

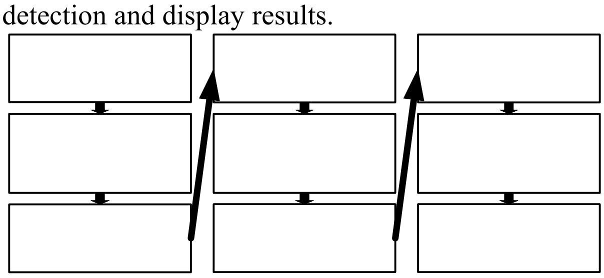

Fig.5 The traffic lane image processing flow chart uses the lane detection algorithm.

The processing results with different conditions are shown in Fig.6, Fig.7 and Fig.8, respectively. Fig.6 exhibits the traffic lane image with clear source and it has been processed in each step of the algorithm. When the input image is without noise addition, it can clearly detect two traffic lanes. Fig.7 presents the algorithm without Gaussian smoothing filter, the traffic lane will be detected long thin one and wrong tiny line. That is not enough to find out whether the vehicles depart the traffic lane. Fig.8 shows the algorithm with generic 2-D Gaussian smoothing filter under the grouping mask 9×11, the result shows both main traffic lanes when vehicle is moving forward like no noise addition case. Fig.9 show the lane detection results in different processing image noise effects. Fig.9(a) exhibits original traffic lane without noise addition. Fig.9(b) shows the traffic lane image with Gaussian noise addition by variance=0.01. Fig.9(c) shows the traffic lane image with generic 2-D Gaussian smooth filter σ=0.9 and grouping mask 9×11. The presenting algorithm can capture two traffic lines under the input image without noise addition while the input image with Gaussian noise addition misses the correct traffic lines. If we add Gaussian smoothing filter on image processing front level, the algorithm can capture the correct traffic line on the both main sides.

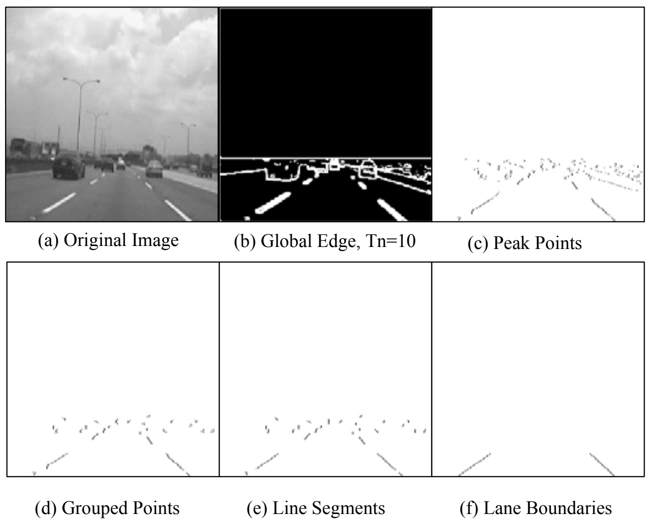

Fig.6 Processing results for lane detection step by step with (V_x, V_y) = (150, 80) and Tn=10.

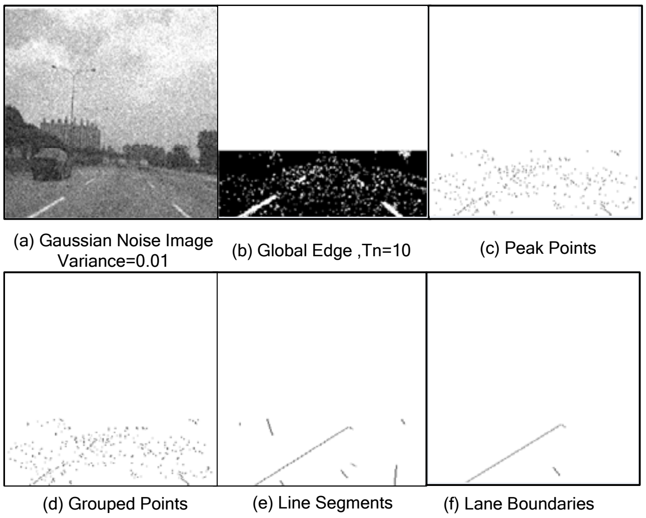

Fig.7 Processing results for lane detection flow and each step images with Gaussian noise, mean=0, variance=0.01 and the grouping mask 9×11.

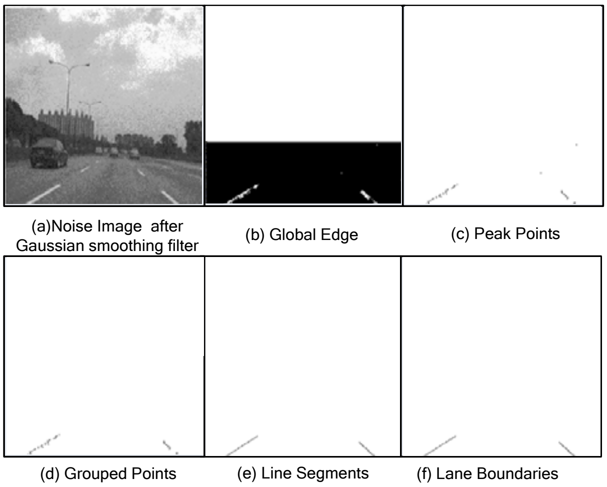

Fig.8 Processing result for lane detection flow and each step images with generic Gaussian 2-D filter and resizing the grouping mask to 9×11.

Fig.9 Lane detection results in different noise addition images. (a) Original traffic lane Image. (b) The traffic lane image with Gaussian noise addition by variance=0.01. (c) The traffic lane image processed by generic 2-D Gaussian smoothing filter with σ=0.9 and the grouping mask 9×11.

Fig.10 shows the experiment platform and the concept drawing of the device placement. The device can be placed under the front windshield of vehicles. The platform is the Pocket PC electronic product from recent market with Xscale 270 CPU.

Fig.10 Result for lane detection using Pocket PC with Xscale270 CPU as the platform

4. Conclusion

In this paper, we present a detailed software handheld lane detection system based on the Pocket PC portable device, which can be easily mounted on real vehicle and dramatically improve safety. The software lane detection algorithm can process the calculation very fast and easily set up on PDA device from recent market. The proposed lane departure detection algorithm let us to detect lane departure under a high vehicle departure speed and avoid false alarm situation successfully. At last, the evaluations prove that this system is robust in most situations including Gaussian noise addition.

Acknowledgement

This work was supported in part by National Science Council, Taiwan, ROC. Under Grant NSC95-2221-E-390-039 and NSC94-2215-E-182 -010.

References:

[1] Young Uk Yim and Se-Young Oh," Three-Feature Based Automatic Lane Detection Algorithm (TFALDA) for Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems.* Vol. 4, No. 4, 2003.
[2] Yuji Otsuka, Shoji Muramatsu, and Tatsuhiko Monji, "Multitype Lane Markers Recognition Using Local Edge Direction," *IEEE Intelligent Vehicle Symposium*, 2002, pp1114-1119.
[3] R. M. Haralick,"Digital Step Edges from Zero Crossing of Second Directional Derivatives", *IEEE Transactions on Panem Analysis and Machine Intelligence*, Vol. PAMI-6, No.1, 1984.
[4] Qing Li, Nanning Zheng, and Hong Cheng, "Springrobot: A Prototype Autonomous Vehicle and Its Algorithms for Lane Detection," *IEEE Transactions on Intelligent Transportation Systems.* Vol. 5, No. 4, 2004.
[5] Jörg Kibbel, Winfried Justus, and Kay Fürstenberg, "Lane Estimation and Departure Warning using Multilayer Laserscanner," *IEEE Conference on Intelligent Transportation Systems*, 2005, pp777-781.
[6] Shih-Shinh Huang, Chung-Jen Chen, Pei-Yung Hsiao, and Li-Chen Fu, "On-Board Vision System for Lane Recognition and Front-Vehicle Detection to Enhance Driver's Awareness," *IEEE International Conference On Robotics And Automation,* Vol. 3, 2004, pp. 2456-2461.
[7] A. L´opez, C. Ca˜nero, J. Serrat, J. Saludes, F. Lumbreras, and T. Graf, "Detection of Lane Markings based on Ridgeness and RANSAC," *IEEE Conference on Intelligent Transportation Systems*, , 2005, pp733-738.
[8] Joel C. McCall and Mohan M. Trivedi, "Video-Based Lane Estimation and Tracking for Driver Assistance: Survey, System, and Evaulation," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 7, No. 1, 2006.
[9] Claudio Rosito Jung and Christian Roberto Kelber, " A Lane Departure Warning System based on a Linear-Parabolic Lane Model," *IEEE Intelligent Vehicles Symposium*, 2004, pp.891 – 895.
[10] J. W. Lee, C.-D. Kee, and U. K. Yi, "A New Approach for Lane Departure Identification," in *Proc. IEEE Intelligent Vehicles Symposium*, Columbus, OH, 2003, pp. 100–105.
[11] Pau-Lo Hsu I, Hsu-Yuan Cheng I, Bol-Yi Tsuei , and Wen-Jing Huang, "The Adaptive Lane-Departure Warning System," *Proceedings of the 41st SICE Annual Conference*, Vol. 5, 2002, pp. 2867-2872.
[12] Claudio Rosito Jung and Christian Roberto Kelber, "A Lane Departure Warning System Using Lateral Offset with Uncalibrated Camera," *IEEE Conference on Intelligent Transportation Systems*, , Sep. 2005, pp348-353.
[13] Pei-Yung Hsiao, and Chun-Wei Yeh, "A Portable Real-Time Lane Departure Warning System based on Embedded Calculating Technique," *IEEE Conference on Vehicular Technology,* 2006. pp.7-04D1-5.
[14] Pei-Yung Hsiao, Chia-Hsiung Chen, Shin-Shian Chou, Le-Tien Li, and Sao-Jie Chen, "A Parameterizable Digital-Approximated 2D Gaussian Smoothing Filter for Edge Detection in Noisy Image," IEEE Int'l Symp. on Circuits and Systems, Island of Kos, Greece, 2006, pp.3189-3192.