

An Image Edge Detection Algorithm Based on Improved Canny

Kewen Liu^{1,2,a}, Kang Xiao^{1,b} and Hongxia Xiong^{3,c}

¹School of Information Engineering, Wuhan University of Technology, Wuhan430070, China

²Key Laboratory of Fiber Optic Sensing Technology and Information Processing, Wuhan University of Technology, Ministry of Education, Wuhan430070, China

³School of Civil Engineering and Architecture, Wuhan University of Technology Wuhan 430070, China

^aliukewen@whut.edu.cn, ^b947919941@qq.com, ^cxionghongxia@126.com

Keywords: Edge detection, Canny operator, Gradient templates, Adaptive filtering.

Abstract. In view of the traditional Canny edge detection algorithm in the smoothing filter will make the image become blurred and threshold selection is not adaptive, and its defects in gradient calculation, this paper proposes an improved Canny edge detection algorithm. The algorithm uses adaptive median filtering instead of Gaussian filtering. The gradient method is adopted to enhance the influence of the middle pixel to improve the edge detection precision. The Otsu algorithm is introduced to calculate the threshold value according to the image pixel information, which avoids the difficulty of artificially setting the high and low thresholds. The simulation results show that the improved Canny algorithm has good anti-noise performance, and it has stronger adaptability when detecting more edge details.

Introduction

Edge is the most basic feature of an image, including the important information of the image, is the premise of the research and analysis of the image[1]. The traditional edge detection algorithm has Roberts operator, Sobel operator, LoG operator and so on. These algorithms are simple and easy to implement, but the ability of denoising is poor and the detection accuracy is not high. In 1986, Canny proposed the Canny edge detection algorithm based on the signal to noise ratio (SNR) criterion, the positioning accuracy criterion and the single response criterion[2]. Compared with the traditional edge detection operator, the Canny edge detection algorithm has been widely used because of its high signal-to-noise ratio and high detection precision. However, the variance of the Gauss function and the choice of the high and low threshold are manually set, adaptive ability is poor. In the literature[3], the threshold value is determined by analyzing the characteristic information of the gradient amplitude histogram, but it can not solve the threshold self-adaptability problem. In the literature[4], the threshold value is obtained by iterative method, but this method has no obvious advantage in dealing with noisy images, and it has a large amount of computation and a long time.

The improved algorithm is improved from three aspects: smoothing filtering, gradient computing[5] and threshold selection. After comparison, the improved algorithm can reflect the main contour features of the image, and achieved good results in the experiment.

Traditional Canny Algorithm

Principle of Algorithm.

The Canny operator is implemented in four parts: image smoothing, calculating the gradient amplitude and direction, Non-maxima suppression, double-threshold detection and connection of edges. The basic flow diagram is shown in Fig.1.

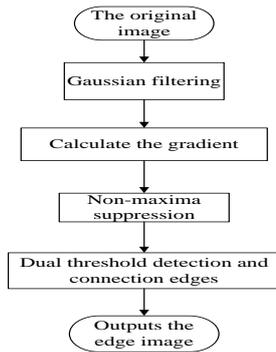


Fig.1 Flow chart of traditional Canny algorithm

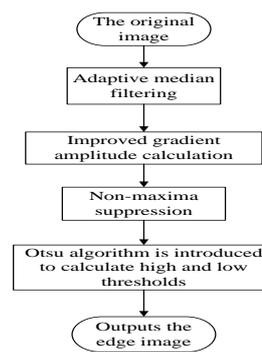


Fig.2 Flow chart of improved Canny algorithm

Realization of Algorithm.

Step1: Gaussian smoothing image

Canny algorithm in the image processing, we must first smooth the original image]to remove noise. The expression of the Gaussian function of the traditional Canny edge detection algorithm is given by the formula (1):

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp(-\frac{x^2 + y^2}{2\sigma^2}) \tag{1}$$

σ is the Gaussian filter parameters. σ smaller, the higher the positioning accuracy, the smaller the signal to noise ratio. σ larger , the lower positioning accuracy, the greater the signal to noise ratio. The choice of σ is very important for edge detection.

Step2: Calculate the gradient magnitude and direction

The gradient magnitude and direction of the smoothed image are obtained by finding the partial derivative. Usually used 2x2 neighborhood of the first-order partial derivative of the image for the x-direction and y-direction difference operation:

$$A_x(i, j) = (I(i, j+1)-I(i, j)+I(i+1, j+1)-I(i+1, j))/2 \tag{2}$$

$$A_y(i, j) = (I(i, j)-I(i+1, j)+I(i, j+1)-I(i+1, j+1))/2 \tag{3}$$

we can calculate the gradient magnitude and direction of the pixel at this time as:

$$A(i, j) = \sqrt{A_x^2(i, j) + A_y^2(i, j)} \tag{4}$$

$$\theta(i, j) = \arctan \frac{A_y(i, j)}{A_x(i, j)} \tag{5}$$

Step3: The non maxima suppression of gradient

In order to better determine the edge, must retain the local maximum gradient point, non maxima suppression[6]. the Canny operator simply compares the gradient values of the target pixel $A(i, j)$ with the gradient amplitude $A_1(i, j)$ and $A_2(i, j)$ of the neighboring pixels on both sides of the gradient direction. If $A(i, j) > A_1(i, j)$ and $A(i, j) > A_2(i, j)$,then $A(i, j)$ remains unchanged. Otherwise, $A(i, j)=0$.

Step4: Double-threshold detection and edge connection

First determine the high and low thresholds T_h and T_l ($T_l=0.6T_h$),if the amplitude of the pixel point gradient is less than the low threshold T_l , it is judged as a non-edge point; if the amplitude of the pixel point gradient is higher than the high threshold T_h , it is judged as the edge point; if the pixel gradient magnitude is between T_l and T_h , It is necessary to determine whether or not the pixel is higher than b in the 8-neighborhood of the pixel, if it exists, it is judged as an edge point, otherwise it is judged as a non-edge point[7].

Improved Canny Edge Detection Algorithm

The improved Canny edge detection algorithm first uses the adaptive median filter instead of the Gaussian filter. Then, the method of enhancing the influence of the middle pixel is adopted to improve the edge detection precision. Finally, the Otsu algorithm is introduced to calculate the high and low thresholds adaptively according to the image pixel information. The algorithm flow chart is shown in Fig.2 above.

Adaptive Median Filtering.

In order to overcome the problem that the variance of Gauss function in traditional Canny algorithm needs to be set manually, we use adaptive median filter instead of Gaussian smoothing to improve the first step of Canny algorithm. Adaptive median filter can remove noise effectively, but also can effectively protect the edge. The basic idea of adaptive median filter is to change the size of the template window, and the signal points and noise points to take a different approach[8], so as to achieve the effect of removing noise.

An image of size $M \times N$, s_{xy} is the working window, s_{max} is the maximum filter window, $f(x, y)$ is the gray value of pixels (x, y) , f_{min} is the minimum pixel gray value, f_{max} is the maximum pixel gray value, f_{med} is the median value.

Adaptive median filtering steps are as follows:

- (1) Initial filter window size $w=3$.
- (2) Calculate f_{min} , f_{max} , and f_{med} in the current template window.
- (3) If $f_{min} \leq f_{med} \leq f_{max}$, then turn (5); otherwise, $w = w + 2$;
- (4) If $w \leq s_{max}$, then turn (2); otherwise, $f(x, y) = f_{med}$.
- (5) If $f_{min} \leq f(x, y) \leq f_{max}$, indicating that the point is not a noise point, the filter output is still $f(x, y)$; otherwise, replace $f(x, y)$ with f_{med} , that is $f(x, y) = f_{med}$.

Improvement of Gradient Magnitude Calculation.

The traditional Canny algorithm is sensitive to noise. In order to overcome the shortcomings of the traditional algorithm for calculating the gradient amplitude, we use finite difference in the 3×2 or 2×3 neighborhood of the enhanced middle pixel weight is used to calculate the gradient, and the template is shown in Fig.3:

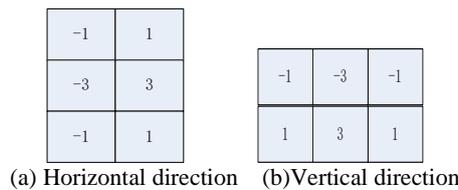


Fig.3 Improved template

The difference between horizontal and vertical directions are:

$$A_x(i, j) = (I(i-1, j+1) - I(i, j) + 3I(i, j+1) - 3I(i, j) + I(i+1, j+1) - I(i+1, j)) / 5 \quad (6)$$

$$A_y(i, j) = (I(i+1, j-1) - I(i, j-1) + 3I(i+1, j) - 3I(i, j) + I(i+1, j+1) - I(i, j+1)) / 5 \quad (7)$$

The gradient amplitude and direction of the pixel at this time can be obtained by the formulas (4), (5), (6) and (7).

Otsu Algorithm Is Introduced to Obtain the Threshold.

The central idea of the Otsu algorithm is that the optimal threshold should be chosen to minimize the intraclass variance and the largest intraclass variance.

Suppose the image to be segmented has N pixels, the gray level is L , The number of pixels is I is n_i , the probability that the pixel gray level is i is $P_i = n_i / N$. The threshold k divides the image into two categories: C_1 and C_2 , C_1 represents the background region, and C_2 represents the target region ($C_1 = [0, 1, 2, \dots, k]$, $C_2 = [k+1, k+2, \dots, L-1]$). the probability that a pixel is assigned to each class is:

$$w_1(k) = \sum_{i=0}^k P_i \quad (8) \quad w_2(k) = \sum_{i=k+1}^{L-1} P_i = 1 - w_1(k) \quad (9)$$

The average grayscale values assigned to pixels of class C_1 and class C_2 are:

$$\mu_1(k) = \sum_{i=0}^k iP(i / C_1) = \sum_{i=0}^k iP_i / \sum_{i=0}^k P_i = \frac{1}{w_1(k)} \sum_{i=0}^k iP_i \quad (10)$$

$$\mu_2(k) = \sum_{i=k+1}^{L-1} iP(i / C_2) = \sum_{i=k+1}^{L-1} iP_i / \sum_{i=k+1}^{L-1} P_i = \frac{1}{w_2(k)} \sum_{i=k+1}^{L-1} iP_i \quad (11)$$

When the gray value is k , the gray cumulative mean:

$$\mu(k) = \sum_{i=0}^k iP_i \quad (12)$$

The gray-level cumulative mean of the entire gray-scale range is:

$$\mu = \sum_{i=0}^{L-1} iP_i \quad (13)$$

From the definition of between-class variance and the above formula, the interclass variance $\sigma_B^2(k)$ is:

$$\begin{aligned} \sigma_B^2(k) &= w_1(k)[\mu_1(k) - \mu]^2 + w_2(k)[\mu_2(k) - \mu]^2 = w_1(k)w_2(k)[\mu_1(k) - \mu_2(k)]^2 \\ &= \frac{[\mu w_1(k) - \mu(k)]^2}{w_1(k)w_2(k)} = \frac{(\sum_{i=0}^k P_i \sum_{i=0}^{L-1} iP_i - \sum_{i=0}^k iP_i)^2}{\sum_{i=0}^k P_i (1 - \sum_{i=0}^k P_i)} \end{aligned} \quad (14)$$

The algorithm traverses the entire gray level of the image. The gray value that is found when σ_B^2 is maximized is the threshold of the optimal segmentation. Let Otsu algorithm obtain the best threshold value for the high threshold T_h , and then according to the low threshold is half the relationship between the high threshold to obtain low threshold T_l .

Simulation Results and Analysis

In order to verify the effectiveness of the improved algorithm, Select the size of 256x256 lena image, by adding noise and without noise to form two images. The experiment is simulated in Matlab2014b environment. Fig.4 is the simulation results of Lena image without noise, Fig.5 is based on the original figure added a mean of 0 and a variance of 0.01 Gaussian noise simulation results.

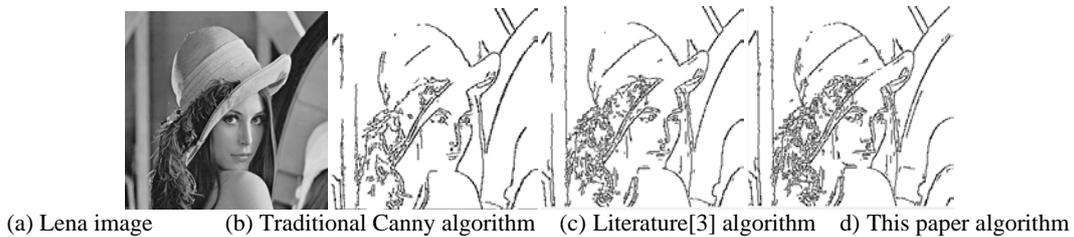


Fig.4 Lena image experiment results

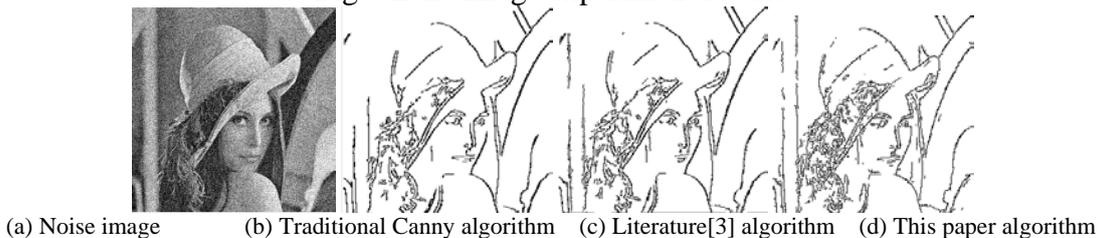


Fig.5 Experimental results of noise - lena image

Observe Fig.4, (b) and (c) are obvious fracture, the edge details are not very good, the face in lena image appears broken, long rod at the left of Fig.4 also appeared obvious fracture. The above situation was significantly better in (d) than (b) and (c), And the edge is more clear, edge positioning is more accurate. The algorithm in this paper is compared with the algorithm in

literature[3], It can be seen from (c) and (d) in Fig.4 that the edges detected by this algorithm are richer, and in terms of edge continuity ,the algorithm is also slightly better than that in the literature[3]. Comparing the simulation results in Fig.5 with the simulation results in Fig.4, it can be seen that the noise has a certain influence on the detection results of these three edge detection algorithms. However, the algorithm of this paper can reduce the influence of noise to the detection effect better. The algorithm can detect more real edges and deal with the details better.

Conclusions

In this paper, we improve the traditional Canny algorithm which is sensitive to noise, poor adaptability and the need to manually set the threshold. The simulation results show that the improved algorithm can detect more edge details, has better effect in noise and edge continuity, and improves the adaptability of the algorithm. It is an effective edge detection algorithm.

Acknowledgments

The author sincerely acknowledges the support of the Key Laboratory of Fiber Optic Sensing Technology and Information Processing, Wuhan University of Technology, Ministry of Education. This research is also sponsored by the Teaching and Research Foundation of Hubei Province (NO.2014110) , as well as the Associate Professor Kewen Liu. The author thanks Zhitao Yang of Wuhan University of Technology. The Corresponding author of this article is Hongxia Xiong.

References

- [1] Gonzalez R C, Woods R E. Digital Image Processing (Third Edition)[M].Beijing Publishing House of Electronics Industry, 2011:445-465.
- [2]Canny J. A Computational Approach to Edge Detection[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 1986, 8(6):679-98.
- [3]Wang S Q,Yao W,Chen J. Edge Detection Based on Histogram Equalization and Morphological Processing [J]. Computer Applications and Software, 2016, 33(3):193-196.
- [4]Deng C X, Wang G B, Yang X R. Image edge detection algorithm based on improved Canny operator[J]. Computer Engineering, 2013, 38(14):168-172.
- [5]Liu L. Improved canny edge detection algorithm matches traffic signs[C]. Eighth International Conference on Digital Image Processing. 2016:100331I.
- [6]Weibo,Yanhui. A New Improved Canny Image Edge Detection Algorithm[J].Imaging Science & Photochemistry, 2016.
- [7]Hong-Ke X U, Qin Y Y, Chen H R. An Improved Algorithm for Edge Detection Based on anny[J]. Infrared Technology, 2014.
- [8]Yang Z D,Zhang X,Zhang T. Improved Adaptive Median Filtering for Image Denoising [J]. Communication Technology, 2015, 48(11):1257-1260.