MATEC Web of Conferences **128**, 01017 (2017) DOI: 10.1051/matecconf/201712801017
EITCE 2017

The Application Study on the Improved Canny Algorithm for Edge Detection in Strain Gauge Image

Qin Wei[1,a], Li Juan[2]

[1]*Shaanxi University of Technology, School of Physics and Telecommunication Engineering, Hanzhong, Shaanxi, 723000, china*
[2]*Shaanxi University of Technology, School of Electrical Engineering, Hanzhong, Shaanxi, 723000, china*

Abstract. In order to realize automatic cutting of strain gauge diaphragm, it is needed to detect and identify strain gauge edge. Though classic canny algorithm can detect the edge of strain gauge, it has such problems as low accuracy, being easily influenced by light, random noise. This paper proposes an improved canny algorithm, replacing Gauss filtering of classic canny algorithm with median filtering, improving gradient magnitude calculation method, and selecting threshold with iteration. Compared to classic canny algorithm the method can improve the ability to resist random noise while detecting strain gauge edge more accurately. Application of the method to strain gauge edge detection will lay foundation for realization of strain gauge automatic cutting.

1 Introduction

Resistance strain gauge is a strain measuring device used most widely and most effective in strain measuring technology of engineering structure. Resistance strain type weighing sensor made of various resistance strain gauge plays vital role in electronic industry. The diaphragm of strain meter in strain gauge production process is 115mm long, 100mm wide, with total 150 strain units. Currently domestic producers still adopt manual cutting, with lower quality, yield and production efficiency than automatic cutting, and strain gauge image edge detection is the key technology that needs to be resolved for strain meter diaphragm automatic cutting. Recently as mathematical theory and AI develop, many new modern edge detection methods emerge, Canny approximately replaces optimal algorithm with first derivative of Gauss function, proposing Canny algorithm, empirically testing effectiveness of the algorithm in detection, positioning and single edge response[1-3]. So far, among existing edge detection algorithms, Canny edge detection is still the most important detection algorithm[4]. But classic canny algorithm has some flaws, such as the problem of response to random noise and fuzzy image boundary. This paper uses median filtering to detect strain gauge image, increasing anti-noise capacity of canny algorithm, meanwhile selects appropriate threshold to eliminate weak edge, increasing accuracy of edge detection.

2 Canny Principle and realization procedure of Canny algorithm

Classic edge detection algorithm realizes edge positioning based on original image, by solving differential of location with obvious gray variation in the image, in brief, uses gradient variation between brightness and darkness. Its biggest advantage is simplicity, and effectiveness. Canny finds that optimal algorithm can be approximately replaced with first derivative of Gaussian function, so he proposes Canny algorithm based on it, and empirically tests the effectiveness of the algorithm in detection, positioning and single edge response. Realization of Canny algorithm mainly includes the following procedures:

(1) Gray processing of source image

Canny algorithm mostly processes gray image, processing color image needs gray processing of image. Gray processing of color gram is to sample red, green, blue channel of image and take weighed mean of sampled value. In terms of RGB format, gray processing often uses the following methods, as shown in equation 1 and 2.

$$Gray = (R+G+B)/3 \quad (1)$$

$$Gray=0.299R+0.587G+0.114B; \quad (2)$$

Where parameters in equation 1 are set mainly considering physical characteristics of human eyes, the color image of other format can also be transformed to RGB format according to corresponding relation before gray processing.

(2) Gaussian filtering processing of image

Filtering operation of image can be achieved by two differential weighing processing of 2 one-dimensional Gaussian cores or one convolution of 1 two-dimensional Gaussian core.

Realization of Gaussian core

$$K = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{x^2}{2\sigma^2}} \quad (3)$$

[a] Corresponding author: Qin Wei@rain301@126.com

DOI: 10.1051/matecconf/201712801017

Equation 3 is discrete form of one-dimensional Gaussian function, where σ is the variance of the function, when the parameter is determined, the one-dimensional core vector is known.

$$K = \frac{1}{2\pi\sigma^2} e^{\frac{x^2+y^2}{2\sigma^2}} \quad (4)$$

Equation 4 is discrete form of two-dimensional Gaussian function, where σ is the variance of the function, when the parameter is determined, the two-dimensional core vector is known.

Gaussian filtering operation of image is essentially a weighing mean operation following specific rules based on gray value of pixel points to be filtered and every point within neighbourhood, which is very useful to filter out high frequency noise mixed in the image.

(3) Calculation of gradient amplitude and direction

Use first differential algorithm to solve partial derivatives of center pixel point in X and Y direction G_x and G_y respectively, and the amplitude and direction of gradient can be obtained, its expressions of first partial derivative, gradient amplitude and direction are shown in equation 5, 6, 7 and 8.

$$G_x = \frac{f[i,j+1] - f[i,j] + f[i+1,j+1] - f[i+1,j]}{2} \quad (5)$$

$$G_y = \frac{f[i,j] - f[i+1,j] + f[i,j+1] - f[i+1,j+1]}{2} \quad (6)$$

$$G = \sqrt{{G_x}^2 + {G_y}^2} \quad (7)$$

$$\theta = arctan\frac{G_y}{G_x} \quad (8)$$

In equation 5, 6, 7 and 8, f[i, j] is center pixel point, G_x and G_y are first partial derivatives of center pixel point, G is its gradient amplitude, θ is direction derivative.

(4) Non-maximum suppression of gradient amplitude

Only global gradient is not enough to determine edge, so in order to determine edge, the point with max local gradient must be retained, and use the direction of gradient to suppress non-maximum, discretize gradient angle into one of four sections of circumference, so as to use a 3 x 3 window to conduct suppression operation. Mark numbers of 4 sections are 0 to 3, corresponding to 4 possible combinations of a 3 x 3 domain, at every point, the center pixel of domain M[x,y] compares with 2 pixels along gradient line. If the gradient value of M[x,y] is not greater than gradient values of two neighbouring pixels, then let M[x,y]=0.

After maximum suppression of image, we will obtain a binary image, but the image still contains many false edges resulting from noise or other causes, so the image needs further processing.

(5) Edge detection and connection of dual-threshold algorithm

The typical method to reduce number of false edge is to use a threshold to N[i, j], giving zero value to all values less than the threshold. Select a threshold using dual-threshold algorithm. Dual-threshold algorithm applies 2 thresholds T_h and T_l to non-maximum suppression image, and $T_h \approx 2T_l$, so 2 threshold edge image $N_1[i,j]$ and $N_2[i,j]$ are obtained. Because $N_2[i,j]$ is obtained using higher threshold, it contains very few false edge, but contains discontinuity (non-closure) [5]. Dual-threshold algorithm will connect edges to contour in $N_2[i,j]$, when arriving at end points of the contour, the algorithm will locate the edge able to connect the contour at the location of 8 neighbouring points of $N_1[i,j]$, as such, the algorithm continuously collects edge in $N_1[i,j]$ till connecting with $N_2[i,j]$. when connecting edges, use array to simulate realization of queue, conducting 8 connected domain search.

3 Improvement of Canny edge detection algorithm

3.1 Improve image filtering method

Median filtering is a nonlinear smoothing technique, setting gray value of every pixel point as mid-value of gray value of all pixel points in a neighbourhood window of the point. Median filtering is based on sequence statistical theory, effectively suppressing noise[6]. The paper uses 3 x 3 two-dimensional rectangle filter. Its operating procedure is: first, rank pixel value in neighbourhood according to gray value. Then, take pixel mid-value of ranked sequence as output mid-value. Finally, compare gray value of surrounding pixel with output mid-value, modify the pixel value with greater difference to a value close to surrounding pixels to realize elimination of noise point[7]. In traditional median filtering, every pixel in filtering zone influences the result equally, without considering the important factor of distance between them and the center, so the proposed median filtering with weighing value distributes an independent weighing value for every location in filtering zone, and distribution of weighing value is decided by the matrix $W(i,j) \in N$. During filtering calculation, every pixel value $I(u+i, v+j)$ is inserted in extended pixel vector, times of insertion id its weighing value $W(i,j)$, extended pixel vector is shown in equation 9.

$$Q = (p_0, \cdots, p_{L-1}) \quad (9)$$

Where, length $L = \sum_{(i,j \in R)} W(i,j)$

Then rank the pixel vector, with a pixel as the center, select a 3 x 3 two-dimensional rectangle neighbourhood, rank the pixel vector according to gray value of every pixel in the neighbourhood. Calculate mid-value y according to equation 10, next further conduct median filtering processing according to mid-value.

$$y = Med\{f_1, \cdots, f_n\} = \begin{cases} x_{i((n+1)/2)} & \text{n is odd} \\ \frac{1}{2}\left(x_{i(n/2)} + x_{i((n+1)/2)}\right) & \text{n is even} \end{cases} \quad (10)$$

Where, y is mid-value, $f_1, \cdots, f_n$ is pixel points in 3 x 3 square neighbourhood, x_i is the value of the array after pixel points in 3 x 3 square neighbourhood are ranked, i is the location of every value in the array, n is the length of the array.

3.2 Improved gradient amplitude calculation method

Canny algorithm calculates gradient amplitude by solving finite difference in 2 x 2 neighbourhood, considering sensitivity of the method to noise, it easily detects false edge, missing real edge. The improved method uses first partial derivative within a 3 x 3 neighbourhood to solve gradient amplitude and direction[8]. First calculate the first partial derivative in the direction of x, y, 45° and 135°, as shown in equation 11, 12, 13 and 14.

X: $F_x(x,y) = I(x+1,y) - I(x-1,y)$ （11）

Y: $F_y(x,y) = I(x,y+1) - I(x,y-1)$ （12）

45°: $F_{45^\circ}(x,y) = I(x-1,y+1) - I(x+1,y-1)$ （13）

135°: $F_{135^\circ}(x,y) = I(x+1,y+1) - I(x-1,y-1)$ （14）

Then, use partial derivatives in the four directions to solve the difference in vertical and horizontal direction, as shown in equation 15, 16:

$$M_H(x,y) = F_x(x,y) + \frac{F_{45^\circ}(x,y) + F_{135^\circ}(x,y)}{2} \quad (15)$$

$$M_V(x,y) = F_y(x,y) + \frac{F_{45^\circ}(x,y) - F_{135^\circ}(x,y)}{2} \quad (16)$$

Finally use equation 17 and 18 to solve gradient amplitude g and gradient direction θ.

$$g = \sqrt{M_V(x,y)^2 + M_H(x,y)^2} \quad (17)$$

$$\theta = arctan\frac{M_V(x,y)}{M_H(x,y)} \quad (18)$$

Where, $F_x(x,y)$、$F_y(x,y)$、$F_{45^\circ}(x,y)$、$F_{135^\circ}(x,y)$ are first partial derivatives in the direction of x, y, 45° and 135°, $I(x,y)$ is the pixel point in 3 x 3 neighbourhood.

3.3 Selection of image segmentation threshold

Threshold segmentation, in brief, is to segment the target of image from background with threshold, and require to minimize the probability of improper segmentation of target from background. When dealing with practical problem, however, due to adverse influence of noise and other inevitable factors on selection of threshold, it is very difficult to determine a threshold meeting requirement. Using iteration algorithm to determine threshold, not only reduces interference of noise and other factors with selection of threshold, but also determines optimal threshold more easily[9].

Mean value T of minimum Z_{min} and maximum Z_{max} of gray value is obtained through histogram statistics and is used as initial threshold, then use the T to divide gray value - part H_0 above T and part H_1 below H, solve mean value TL of H_1, then again solve mean value of T_H and T_L as threshold, and the threshold is new threshold obtained after our first iteration, compare obtained new threshold with the initial threshold. It their value is the same or their difference meets the condition we set, the iteration ends, T_H and T_L are required optimal high, low threshold. If condition is not met, use the threshold obtained this time to divide all gray value again, continue the process of iteration till new threshold meets requirement[10]. New threshold calculated through multiple iteration will be much better than initial threshold, minimizing probability of improper segmentation of image. Specific procedure of iteration algorithm is as follows:

(1) Make statistic of histogram of image, use equation 19, 20 to calculate initial threshold T of image

$$T = \{T_k | k = 0\} \quad (19)$$

$$T = \frac{Z_{min} + Z_{max}}{2} \quad (20)$$

Where K is times of iteration, Z_{max} represents max gray value, Z_{min} represents min gray value.

(2) Divide image into part H_0 above T and part H_1 below H, as shown in equation 21, 22.

$$H_0 = \{f(i,j) | f(i,j) \geq T\} \quad (21)$$

$$H_1 = \{f(i,j) | f(i,j) < T\} \quad (22)$$

(3) Calculate gray mean value T_H ,T_L of H_0 and H_1, as show by equation 23, 24.

$$T_H = \frac{\sum_{f(i,j)\geq T} f(i,j)}{\sum_{f(i,j)\geq T} N_0(i,j)} \quad (23)$$

$$T_L = \frac{\sum_{f(i,j)<T} f(i,j)}{\sum_{f(i,j)<T} N_1(i,j)} \quad (24)$$

Where $f(i,j)$ is gray value of pixel point (i,j), $N_0(i,j)$, $N_1(i,j)$ meet conditions of equation 25, 26 respectively.

$$N_0(i,j) = \begin{cases} 1, f(i,j) \geq T \\ 0, \quad \text{else} \end{cases} \quad (25)$$

$$N_1(i,j) = \begin{cases} 1, f(i,j) < T \\ 0, \quad \text{else} \end{cases} \quad (26)$$

(4) Calculate mean value of T_H and T_L as new threshold TT, as shown in equation 27.

$$TT = \frac{T_H + T_L}{2} \quad (27)$$

(5) If TT and initial threshold have same value or meet the condition we set, then iteration is completed, if TT obtained this time fails to meet requirement, then continue iteration till TT meeting requirement is found.

(6) Iteration completes, use the result of calculation TT - T_H and T_L meeting condition as optimal high, low threshold for detection.

4 Result of strain gauge image edge detection

4.1 Experimental test of edge detection method

The key for Canny algorithm to detect image edge is selection of Gaussian filtering function σ and threshold. Fig. 1 a is the image of original strain gauge, Fig. 1 b is the image of edge detected by classic canny algorithm, Fig. 1 c Image of filtering and gradient solution improved Canny algorithm edge detection, Fig. 1 d Image of threshold obtaining method improved Canny algorithm edge detection

To test processing effect of improved canny algorithm on Gaussian noise, add Gaussian noise to strain gauge original image. Fig. 2 a is Gaussian noise added original image, Fig. 2 b is classic canny algorithm detected edge image, Fig. 2 c is filtering and gradient amplitude solution improved canny algorithm detected edge image, Fig. 2 d is threshold improved canny algorithm detected edge image.

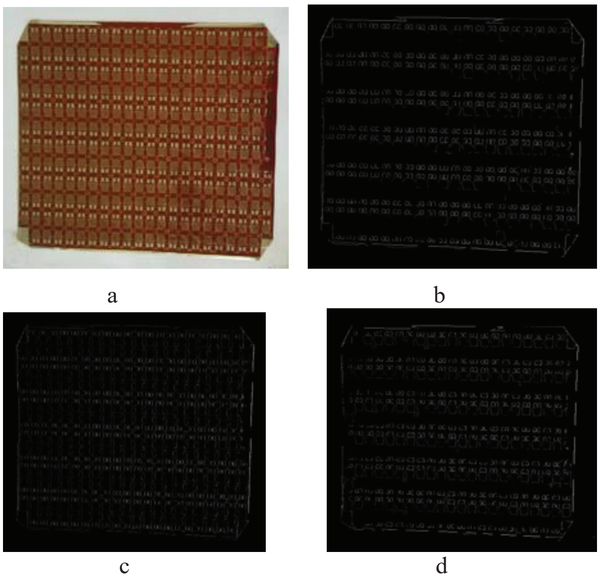

Figure 1. Result of detection without noise edge

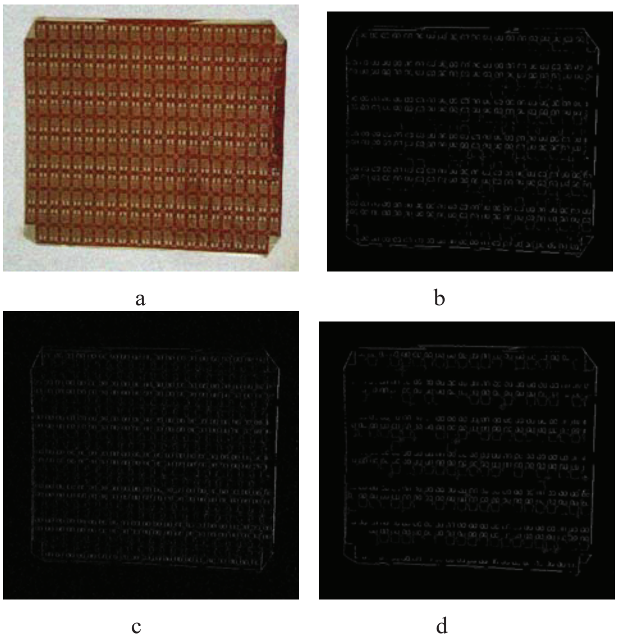

Figure 2. Result of edge detection when adding Gaussian noise

4.2 Analysis of experimental result

1. Without adding noise, compare edge detection results of different Canny algorithm in Fig. 1, it can be found that compared to edge characteristics detected by classic Canny algorithm, threshold obtaining method improved algorithm detect more real edges when automatically selecting threshold, and filtering and gradient solution improved algorithm detects most complete edge characteristics among three algorithms when selecting very good threshold.

2. When adding Gaussian noise, compare detection results of Fig. 2, it can be found that the algorithm using Gaussian filtering almost filter out all false edges produced by noise, but it also filters out many real edges, especially when using traditional algorithm of manually setting threshold, and the algorithm using median filtering keeps most real edges though, it also keeps many false edges resulting from noise.

5 Conclusion

There are many image edge detection methods, but precision and time of strain gauge cutting need to be guaranteed during strain gauge edge detection. Aiming at these requirements, based on classic canny algorithm, during canny algorithm edge detection processing, the design optimizes Gaussian filtering processing and threshold selection. Experimental result shows that the optimization method proposed in the paper has obvious effect on increasing accuracy of strain gauge edge detection and anti-noise capability, as well as certain reference value for application of strain gauge visual detection cutting.

Acknowledgment

The Special Scientific Research project of Education Department of Shaanxi Province(NO.17JK0139).

References

1. Li Hong Dong; Peng Bing Zhao.Application of Improved Canny Edge Detection Algorithm in Coal-Rock Interface Recognition[J].Applied Mechanics and Materials. 1279-1283(2012).
2. Ranita Biswas; Jaya Sil.An Improved Canny Edge Detection Algorithm Based on Type-2 Fuzzy Sets[J].Procedia Technology.820-824(2012).
3. Mingjie Wang; Jesse S. Jin; Yifei Jing;Xianfeng Han;Lei Gao;Liping Xiao.The Improved Canny Edge Detection Algorithm Based on an Anisotropic and Genetic Algorithm[J].Advances in Image and Graphics Technologies.115-124(2016).
4. Tao Sun; Chang Zhi Gao.An Improved Canny Edge Detection Algorithm[J].Applied Mechanics and Materials.2869-2873(2013).
5. Yupeng Wang; Jiangyun Li.An Improved Canny Algorithm with Adaptive Threshold Selection[J].MATEC Web of Conferences.(2015)
6. Alejandro Díaz-Sánchez; Jaime Ramírez-Angulo;Javier Lemus-Lopez.Analog Adaptive Median Filters[J].Analog Integrated Circuits and Signal Processing. **36**,3,207-213(2003).
7. A. I. Kravchonok (1); B. A. Zalesky (1);P. V. Lukashevich (1).An algorithm for median filtering on the basis of merging of ordered columns[J].Pattern recognition and image analysis. **17**,3,402-407(2007).
8. Yang, Tao; Qiu, Yue-hong.Improvement and implementation for Canny edge detection algorithm[J].Proceedings of the SPIE.(2015)
9. Dan Dan Han; Tian Chi Zhang;Jing Zhang.Research and Implementation of an Improved Canny Edge Detection Algorithm[J].Key Engineering Materials.566-569(2013).

DOI: 10.1051/matecconf/201712801017

10. Wei Qing Wang; Hua Feng Ding;Xing Jiang Xiao;Jun Xu;Yuan Zhang.The Improved Algorithm of Edge Detection Based on Canny Operator[J].Advanced Materials Research.97-102(2011).