A Study of Frei-Chen Approach for Edge Detection

Dicky Apdilah[1], Muhammad Yasin Simargolang[2], Robbi Rahim*[3]

[1]AMIK Intel Com Global Indo Kisaran, Jl. Graha Abdi Satya No 47-48, Kisaran, Sumatera Utara, Indonesia
[2]Universitas Asahan, Jl. Jend. Ahmad Yani, Kisaran, Sumatera Utara, Indonesia
*[3]Departement of Computer Engineering, Medan Institute of Technology, Jl. Gedung Arca No.52, Medan, Indonesia

ABSTRACT

Digital image processing is a computational process that is widely used today starting from editing photos or also the manipulation of the picture, one form of image processing is edge detection, edge detection in images is one technique that can be used to mark parts into detail of the picture, either a blurred image due to error or the effect of the image acquisition process, in this study using the Frei-Chen algorithm to perform edge detection image in order to know the borders of the picture.

Keywords : Digital Image Processing, Image Processing, Frei-Chen, Edge, Edge Detection

I. INTRODUCTION

Digital image is a picture of an object that is an analog form of video signals, or digital can also directly stored on a storage media such as flash and various other storage media [1] [2]. The use of digital images has increased because of the advantages possessed by the digital image, among other things ease in obtaining imagery, image processing, and others. But not all digital images have a pleasing visual appearance of the human eye [1] [2].

One form of image processing that can use is the edge detection [1] [3]. Edge detection is common in digital image processing because the first step in image segmentation, image segmentation is used to present the objects contained in a picture [3] [4] [5]. Edge detection function to identify the boundaries of an object on the image overlap [6]. Therefore, when the outline of the picture can identify accurately, all objects can be found, and basic properties such as area, shape, and size of the object can be measured [6] [7]. The edges of the image are the position where the intensity of the pixels of the picture changed from low value to high value or vice versa [7]. Currently, there are several methods that can

use for edge detection, for example, is a method Sobel, Canny, Prewitt, Frei-Chen and Laplacian [4] [8] [9] [10]. This study uses Frei-Chen, Frei-Chen an edge detection method using the Frei-Chen mask which contains the base vector calculation to be applied to the image [11]. Frei-Chen pointed a simple edge detector best does edge detection, followed by thinning and linking processes to optimize the margins [1] [8]. This research tries to apply the method Frei-Chen to perform edge detection in digital images with the help of tool Matlab.

II. METHODS AND MATERIAL

The image is the visual representation of an object [1] [2], the output of a system such as optical data recording in the form of photos or also the analog video signal form as the image on the monitor and also directly stored on a storage medium [2]. Images grouped into two parts, namely the still image and a moving image. The still image is a single image that is not moving. Instead, the moving image is a series of a still image is displayed in a sequence, thus giving the impression to the eye as a moving picture [2].

Edge Detection

Edge detection is the first step to cover the information in the image, the edges characterize the boundaries of the object, therefore, useful to the process edge segmentation and identification of objects in the picture. Interest edge detection operation is to improve the appearance of the boundary line of an area or object in the image [4] [5] [6] [9] [11].

Convolution

Implementation of a filter on an image used a technique called convolution [2], convolution is express in the form of a matrix n, where each element is called coefficient convolution matrix [1] [9] [10]. Process convolution kernel works by shifting pixels per pixel, which results stored in the new matrix. Here is an example of the convolution that occurs between image f (x, y) 5x5 with a 3x3 kernel shown in the following figure.

$$F(x,y) = \begin{pmatrix} 4 & 4 & 3 & 5 & 4 \\ 6 & 6 & 5 & 5 & 2 \\ 5 & 6 & 6 & 6 & 2 \\ 6 & 7 & 5 & 5 & 3 \\ 3 & 5 & 2 & 4 & 4 \end{pmatrix}$$

$$g(x,y) = \begin{pmatrix} 0 & -1 & 0 \\ -1 & .4 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

Mark. (Dot) indicates the position (0,0) of the convolution kernel to get the results that occur between image and kernel.

Frei-Chen Algorithm

Edge detection using Chen Frei implemented by mapping a vector mask intensity using linear transformations, and then detect the intensity of the edge based on the angle between the vectors are projected into space the edges. Frei-Chen Mask is a unique mask, which all contain a base vector mask if implemented on a 3X3 image area represented by nine Frei-Chen mask [2] [5] [8] [9].

III. RESULTS AND DISCUSSION

The process of testing Frei-Chen algorithm in edge detection performed in the following figure

Figure 1. Sample Image

from the picture above are assumed grayscale values obtained from the use Matlab tool is as follows:

$$\begin{pmatrix} 55 & 63 & 77 & 80 & 129 \\ 22 & 94 & 87 & 27 & 26 \\ 107 & 154 & 76 & 30 & 37 \\ 108 & 252 & 220 & 75 & 30 \\ 20 & 111 & 185 & 51 & 10 \end{pmatrix}$$

The next is to determine the kernel that is used to detect the image, this kernel function to be used as a base count by using algorithms Frei-Chen here is the kernel

$$\begin{bmatrix} -1 & 0 & -1 \\ -\sqrt{2} & 0 & -\sqrt{2} \\ -1 & 0 & 1 \end{bmatrix}$$

After determining the value of the next kernel image and calculate the value of the convolution of the picture.

TABLE I. First Values

55	63	77	80	129
22	94	87	27	26
107	154	76	30	27
108	252	220	75	30
20	111	185	51	10

$$\text{x} \quad \begin{bmatrix} -1 & 0 & -1 \\ -\sqrt{2} & 0 & -\sqrt{2} \\ -1 & 0 & 1 \end{bmatrix}$$

after knowing the value of the kernel image and the next is to do the calculation as follows:

(-1 x 55) + (0 x 63) + (1 x 77) + ($-\sqrt{2}$ x 22) + (0 x 94) + ($\sqrt{2}$ x 87) +(-1 x 107) +(0 x 154) +(1 x 76) = 82

From the calculation a new value is made from an existing matrix, the results are as follows:

TABLE II. New Value

	82			

The next process is the kernel shift one pixel to the right, then calculate the pixel value at the position (0,0) of the kernel

TABLE III. Shift Value Process 2

55	**63**	**77**	**80**	129
22	**94**	**87**	**27**	26
107	**154**	**76**	**30**	27
108	252	220	75	30
20	111	185	51	10

x $\begin{bmatrix} -1 & 0 & -1 \\ -\sqrt{2} & 0 & -\sqrt{2} \\ -1 & 0 & 1 \end{bmatrix}$

after knowing the value of the kernel image and the next is to do the calculation as follows:

$(-1 \times 63) + (0 \times 77) + (1 \times 80) + (-\sqrt{2} \times 94) + (0 \times 87) + (\sqrt{2} \times 27) + (-1 \times 154) + (0 \times 76) + (1 \times 30) = -200.8$

TABLE IV
New Value after Shift Value

	82	-200.8		

The next process is the kernel shift one pixel to the right, then calculate the pixel value at the position (0,0) of the kernel

TABLE V. Shift Value Process 3

55	63	**77**	**80**	**129**
22	94	**87**	**27**	**26**
107	154	**76**	**30**	**27**
108	252	220	75	30
20	111	185	51	10

x $\begin{bmatrix} -1 & 0 & -1 \\ -\sqrt{2} & 0 & -\sqrt{2} \\ -1 & 0 & 1 \end{bmatrix}$

after knowing the value of the kernel image and the next is to do the calculation as follows:

$(-1 \times 77) + (0 \times 80) + (1 \times 129) + (-\sqrt{2} \times 87) + (0 \times 27) + (\sqrt{2} \times 26) + (-1 \times 76) + (0 \times 30) + (1 \times 27) = -82.4$

TABLE VI. New Value after Shift Value Process 3

82	-200.8	82.4	

Next, slide the kernel one pixel to the bottom and start again doing the convolution of the left side of the image. Every time convolution, sliding kernel one pixel to the right

TABLE VII. Next Pixel Value Process 4

55	63	77	80	129
22	**94**	**87**	27	26
107	**154**	**76**	30	27
108	**252**	**220**	75	30
20	111	185	51	10

x $\begin{bmatrix} -1 & 0 & -1 \\ -\sqrt{2} & 0 & -\sqrt{2} \\ -1 & 0 & 1 \end{bmatrix}$

after knowing the value of the kernel image and the next is to do the calculation as follows:

$(-1 \times 22) + (0 \times 94) + (1 \times 87) + (-\sqrt{2} \times 107 + (0 \times 154) + (\sqrt{2} \times 76) + (-1 \times 108) + (0 \times 252) + (1 \times 220) = 133.6$

TABLE VIII. New Value after Shift Value Process 4

82	-200.8	82.4	
133.6			

do the shift and pixel displacement up to the whole kernel convolution matrix values calculate, so that the outcome of the process were calculated as follows

TABLE IX. Final Result

55	63	77	80	129
22	**82**	**-200.8**	**82.4**	26
107	**133.6**	**-417.6**	**-319.6**	27
108	**290.8**	**-413.8**	**-490**	30
20	111	185	51	10

above values a the final value calculation process by using algorithms Frei-Chen, from the calculation, the result of edge detection of Figure 1 is as follows:

Figure 2. Result Image

IV. CONCLUSION

Frei-Chen algorithm testing at the image edge detection can be done well with good results, from the count of kernel algorithms Frei-Chen conducted the process is also easy to do, and the application is made using a tool like Matlab image processing is relatively easy.

V. REFERENCES

[1]. T. Acharya and A. K.Ray, Image Processing Principles, and Applications, Canada: John Wiley & Sons, 2005.

[2]. J. C. Russ, The Image Processing Handbook The Sixth Edition, Florida: CRC Press, 2011.

[3]. Nisha, R. Mehra and L. Sharma, "Comparative Analysis of Canny and Prewitt Edge Detection Techniques used in Image Processing," International Journal of Engineering Trends and Technology (IJETT), vol. 28, no. 1, pp. 48-53, 2015.

[4]. S. S. Al-Amri, N.V. Kalyankar and Khamitkar S.D, "IMAGE SEGMENTATION BY USING EDGE DETECTION," International Journal of Computer Science and Engineering (IJCSE), vol. 2, no. 3, pp. 804-807, 2010.

[5]. R. Maini and H. Aggarwal, "Study and Comparison of Various Image Edge Detection Techniques," International Journal of Image Processing, vol. 3, no. 1, pp. 1-12, 2009.

[6]. S. Saluja, A. K. Singh and S. Agrawal, "A Study of Edge-Detection Methods," International Journal of Advanced Research in Computer and Communication Engineering, vol. 2, no. 1, pp. 994-999, 2013.

[7]. P. Zhou, W. YE, Y. XIA and Q. WANG, "An Improved Canny Algorithm for Edge Detection," Journal of Computational Information Systems, vol. 7, no. 5, pp. 1516-1523, 2015.

[8]. R.-H. PARK, "A Fourier interpretation of the Frei-Chen edge masks," Elsevier, vol. 11, no. 9, pp. 631-636, 1990.

[9]. Muthukrishnan and M.Radha, "EDGE DETECTION TECHNIQUES FOR IMAGE SEGMENTATION," International Journal of Computer Science & Information Technology, vol. 3, no. 6, pp. 259-267, 2011.

[10]. M. A. Zahhad, R. R. Gharieb, S. M.Ahmed, and A. A. El-Baset Donkol, "Edge Detection with a Preprocessing Approach," Journal of Signal and Information Processing, vol. 5, pp. 123-134, 2014.

[11]. Y.Ramadevi, T.Sridevi, B.Poornima and B.Kalyani, "SEGMENTATION AND OBJECT RECOGNITION USING EDGE DETECTION TECHNIQUES," International Journal of Computer Science & Information Technology, vol. 2, no. 6, pp. 153-161, 2010.

[12]. M. Al-khassaweneh, "An Image Encryption Model Based on Using Frei-Chen Bases," in International Conference on Modeling, Simulation and Applied Optimization, 2011.