http://infor.seaninstitute.org/index.php/infokum/index
JURNAL INFOKUM, Volume 8, No.2, JUNI 2020 ISSN 2302-9706

IMAGE EDGE DETECTION USING ROBINSON OPERATOR

Pandi Barita Nauli Simangunsong
Program Studi Komputerisasi Akuntansi
AMIK Imelda, Medan, Sumatera Utara, Indonesia
Pandibarita45@gmail.com

Abstract

Image processing is now widely used in almost all fields including medicine, industry, agriculture, geology, marine, and so on. One of the main things in image processing is the process of detecting the edge of an image, wherewith this process the boundary edge of an object with its background can be well determined. Edge detection is a method that can be used to detect the edge of an image that aims to improve the appearance of the boundaries of an area or objects in the image so that objects or boundaries of an area in the image are more easily recognized by humans and machines. Edge detection in digital images aims to recognize a pattern contained in the image itself. With the recognition of patterns in an image will be easily obtained information in an image. Determining the location of the edge of an image is easy if the condition of the image is clear and sharp, but the accuracy of the existence of the edge becomes difficult to determine if there is interference in the image, such as noise. But often it also results in an unsustainable image border where the results are influenced by the type of method used.
Abstract: Robinson, detection, image

1. Introduction

Image processing is now widely used in almost all fields including medicine, industry, agriculture, geology, marine, and so on. One of the main things in image processing is the process of detecting the edge of an image, wherewith this process the boundary edge of an object with its background can be well determined. Edge detection is a method that can be used to detect the edge of an image that aims to improve the appearance of the boundaries of an area or objects in the image so that objects or boundaries of an area in the image are more easily recognized by humans and machines.

The basis of the edge detection technique or in English is called edge detection is to search the image vertically or horizontally while looking at whether there is a color change that exceeds a sensitivity between two adjacent points. If there is a change, then between the two points is considered the edge (edge) of an image. Detection of the image can be done by several methods including using the methods of Robert's Operator, Prewitt Operator, Sobel Operator, Kirsch Operator, and Robinson Operator 3 Level.

According to Ayu Leonitami, One method that can be used in edge detection is to use the Robinson operator 3 level method. Robinson operator detection 3 levels not only performs the process of detecting edges from the vertical and horizontal directions, just like Robert's operator method and Sobel operators are more sensitive to diagonal edges, whereas Robinson operator 3 levels search for edges using eight cardinal directions, as does the detection process the Prewitt edge and the kirsch operator differ only in the kernel (mask) value. The process is to convolution the grayscale image using eight kernels (masks) and the results of the operation are the maximum values of the eight convolutions. Various edge detection techniques work in different ways. Each has strengths, which is why, experiments in an application using various edge detection techniques need to be done to get the best results. [1], [2]

2. Literature Rivew

2.1 Image Definition

Image is a representation of similarity, or imitation of an object. Image as the output of a data recording system can be optical in the form of photos, analogous in the form of video signals such as images

on television monitors, or digital that can be directly stored on a storage medium. The concept of an image is a function is an important concept in the process of developing and implementing image processing techniques. [3]–[5]

2.2 Image Representation

Digital images are formed by a collection of dots called pixels (pixels or picture elements). Each pixel is described as a small square. Each pixel has position coordinates. The coordinate system used to express digital images is shown in the image below

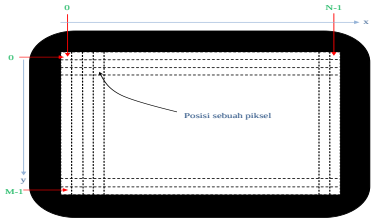

Figure 1. Image Coordinate System with MxN Size

With the coordinate system following the scanning principle on the standard TV screen, a pixel has coordinates in the form of (x, y). In this case x represents the column position and y represents the row position, the upper-left corner pixel has coordinates (0, 0) and the pixels in the lower-right corner have coordinates (N-1, M-1). [6]–[8]

2.3 Image Resolution

In addition to counting the brightness intensity, the number of pixels used to compile an image affects the quality of the image. The term ordinary image resolution is stated by the number of pixels in the width and height direction. The usual pixel resolution is expressed by the notation m x n, with m expressing height and n expressing width in the number of pixels. The example in Figure 2.7 shows that if an apple image is only expressed in 8 x 8 pixels, the image formed is very different from the original. If the number of pixels used is more, of course, it will be closer to the original image [9]

2.4 Robert Operator

Operator Roberts, first published in 1965, consists of two 2x2 filters. The small filter size makes computing very fast. However, these advantages at the same time cause weaknesses, which are greatly affected by noise. In addition, the Roberts operator gives a weak response to edges, unless the edges are very sharp.

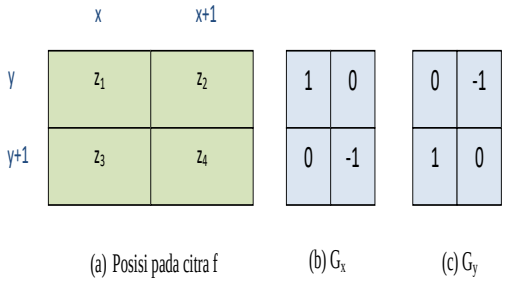

Figure 2. **Operator Roberts**

The form of the Roberts operator is shown in Figure 2.40. For example, f is the image that will be subject to the Roberts operator. Therefore, the Roberts operator value at (y, x) is defined as [10]–[12]

$$r(y,x) = \sqrt{(z_1 - z_4)^2 + (z_3 - z_2)^2}$$

wher , z_1 = f(y, x), z_2 = f(y, x+1), z_3 = f(y+1, x), dan z_4 = f(y+1, x+1).

3. Method

Robinson operator work process by following the following flowchart:

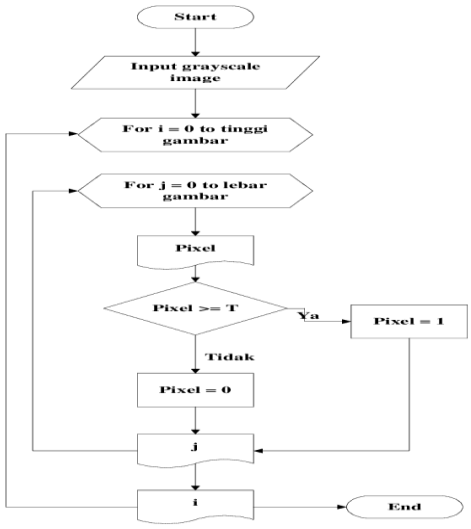

Figure 3. Robinson Operator Flowchart

4. Analysis and Discussion

4.2 Input Analysis

Input image in the form of an image that has a color intensity ranging from 200 x 150 x 3 = 90000 color intensity or commonly called RGB, which is a type of image that presents colors in the form of components R (red), G (green), B (blue). Each color component uses 8 bits (values range from 0 to 255).

4.3 Process.

The process in this system processes grayscale images to be edge detection on images after inputting images processes RGB images into grayscale images, the system will convolution grayscale images to detect edges on digital images using the 3-level Robinson operator method. Robinson operator 3 levels are used to detect all edges from various directions in the image. Robinson operator 3 levels used for edge detection display the edges of various eight cardinal directions such as from north, northeast, east, southeast, south, southwest, west, and northwest. Edge detection is done by using image convolution with various 3 level Robinson operator masks, then the greatest magnitude of the edge strength and direction is sought. If for example z used 3-level Robinson operator masks and edge strength values in pixels (y, x) for all masks are G1 [f (y, x)], G2 [f (y, x), ... , Gp [f (y, x)], then the edge strength is:

$h_{n,m}=\max_{z=1,\ldots,8} \sum_{i=-1}^{1} \sum_{j=-1}^{1} G_{ij}^{(z)}.f_{n+i,m+j}$

if max z is the mask that gives the greatest strength, then the edge direction is determined from the z mask. Because a 3-level operator Robinson convolution table, calculating the direction and presence of an edge is done in three main steps, namely:

1. Calculation of derivatives for each of the eight directions. The equation for derivatives is written in the terminology of the elements of a 3x3 table.

Step I: Robinson operator 3 level mask convolution process uses a mask from the east and uses a 3x3 image as below:

$$f(y,x)=\begin{bmatrix} 85 & 85 & 85 \\ 85 & 85 & 85 \\ 85 & 85 & 85 \end{bmatrix} \qquad g(y,x)=\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

3x3 Image Pixel Value, Eastern level Robinson Mask 3 operator, Eastern operator Robinson mask convolution process uses equations that have been explained based on the theory. Examples of calculations can be done as follows:

Deriv_T = 1 x (85+85+85) – 1 x (85+85+85)
= 255 - 255
= 0

Step III: The Robinson operator 3 Level mask convolution process uses a mask from the north and uses a 3x3 image as below:

$$f(y,x)=\begin{bmatrix} 8 & 8 & 8 \\ 85 & 8 & 8 \\ 8 & 8 & 85 \end{bmatrix} * g(y,x)=\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}$$

3x3 Pixel Image Value of Robinson Operator Mask 3 North Level The process of convolution of Robinson operator 3 Level north uses an equation that has been explained based on the theory. Examples of calculations can be done as follows:

Deriv_U = 1 x (85+85) – 1 x (85+85+85)
= 170 - 255
= -85

Step IV: The 3 Level robinson operator convolution process uses a mask from the northwest and uses a 3x3 image as below:

$$f(y,x)=\begin{bmatrix} 85 & 85 & 85 \\ 85 & 85 & 85 \\ 85 & 85 & 85 \end{bmatrix} * g(y,x)=\begin{bmatrix} 1 & 1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & -1 \end{bmatrix}$$

3x3 Pixel Value of the Northwest Operator Robinson Mask Convolution process of the northwest robinson mask operator uses an equation that has been explained on the basis of the theory. Examples of calculations can be done as follows:

Deriv_B = 1 x (85+85+85) – 1 x (85+85+85)
= 255 - 255
= 0

Step VI: The robinson operator mask convolution process uses a mask from the southwest and uses a 3x3 image as below:

$$f(y,x)=\begin{bmatrix} 85 & 85 & 85 \\ 85 & 85 & 85 \\ 85 & 85 & 85 \end{bmatrix} * g(y,x)=\begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix}$$

Image Pixel Value of 3xx, Southwestern Robinson Mask Mask, Southwestern robinson operator convolution process uses equations that have been explained on the basis of theory. Examples of calculations can be done as follows:

Deriv_BD = 1 x (85+85+85) – 1 x (85+85+85)
= 255 - 255
= 0

From the results of the convolution of the Robinson operator 3 Level method of the 3x3 grayscale image using the 3x3 mask as explained above, the maximum value of the eight masks from the direction:

East = 0
Northeast = 0
North = -85
Northwest = 0
West = 0

Southwest = 0
South = 0
Southeast = 0
Then the maximum value from any direction all the results are intensity 0. After knowing the maximum value and direction, then the maximum value from any direction will be edge detection. With the greatest intensity value, if the intensity value is the same then, the one that is processed is the first time it is calculated. The results of the convolution of grayscale images with the robinson operator 3 Level method can be seen in the table

Table 1. Intensity Value of Convolution Results

Y\X	*1*	*2*	*3*	*4*	*5*	*198*
1	0	0	0	0	0	
2	0	0	0	0	0	
3	0	0	0	0	0	
4	0	0	0	0	0	
5	1	5	6	5	0	
148	.	.	.	.	.	.

In table 2 can be seen f (y, x) row 1 and column 1 the intensity value is 0 which is the intensity value of the grayscale image in row 2 and column 2 whose intensity value is 85 in table 3 so the intensity value of 85 is exchanged to 0, in row first, last row, first column and last column are not processed because they do not have complete neighbors (as many as 8 neighbors) and in table 3.5 the image size is reduced to 148 x 198 because in the first row, the first column, the last row and the last column are removed because they are not processed .

4.3 Image analysis Results

The resulting image is in the form of the 3 level robinson operator convolution process which is the final image.

Figure 5. Robinson operator results

5. Conclusions

Image processing convolution process with edge detection using Robinson Operator Level 3 method can process grayscale images by utilizing the eight neighbors and can determine the edge direction of the image with eight kernels (masks) and draw black entry into white as the largest magnitude direction of the eight Robinson kernels 3 Level Operators.

Reference

[1] N. Nafi'iyah, "Algoritma Kohonen dalam Mengubah Citra Graylevel Menjadi Citra Biner," *J. Ilm. Teknol. Inf. Asia*, 2015.
[2] A. Wedianto, H. L. Sari, and Y. S. H, "Analisa Perbandingan Metode Filter Gaussian , Mean Dan Median Terhadap Reduksi Noise," *J. Media Infotama*, 2016.
[3] R. Munir, "Pengantar Pengolahan Citra," *Pengolah. Citra Digit.*, 2013.

[4] R. Munir, "Citra Biner," *Pengolah. Citra Digit. Dengan Pendekatan Algoritm.*, 2005.
[5] N. Ahmad and A. Hadinegoro, "Metode Histogram Equalization untuk Perbaikan Citra Digital," *Semin. Nas. Teknol. Inf. Komun. Terap.*, 2012.
[6] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.*, 2010.
[7] N. Wang and D. Y. Yeung, "Learning a deep compact image representation for visual tracking," in *Advances in Neural Information Processing Systems*, 2013.
[8] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015.
[9] C. Dong, C. C. Loy, K. He, and X. Tang, "Image Super-Resolution Using Deep Convolutional Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2016.
[10] B. Ilschner and R. F. Singer, *Werkstoffwissenschaften und Fertigungstechnik*. 2016.
[11] R. Eriksson, N. Modig, and K. Elkington, "Synthetic inertia versus fast frequency response: A definition," in *IET Renewable Power Generation*, 2018.
[12] H. Atlam, R. Walters, and G. Wills, "Fog Computing and the Internet of Things: A Review," *Big Data Cogn. Comput.*, 2018.