VLSI Implementation of Hough Transform To Reduce Resource on an Image

T.M.Senthil Ganesan[1], Mr.T.Siva Kumar[2]
[1]Assistant Professor, Department of CSE, Velammal College of Engineering & Technology, Madurai
[2]Assistant Professor, Department of ECE, Fatima Michael College of engineering and technology, Madurai

Abstract-*In the circle detection process, hough transform techniques evaluated to determine if they are part of a candidate circle and center and radius of the circle is computed. The algorithm has been tested on a images of using Matlab. Edge images of tires or coins are compared with images obtained using canny edge detection. Useful results for detecting circles in images are found. VHDL implementation of the extraction algorithm is discussed. The results indicate that the design code can perform evaluations memory usage is small compared to that required for the circular Hough transform. The time taken to the center and found circle, and memory usage reduced by the proposed algorithm.*

Keywords*-*VHDL, Hough transform.

I. INTRODUCTION

Detecting and recognizing the shapes in an image is extremely important in industrial applications in recognizing the object. Detecting circles in an image is one of the problems that is discussed in this thesis. Many algorithms, such as Linear Square Method and Hough Transform, have been proposed to detect circles. These algorithms detect circles from the edge detected images. Among these algorithms, Circular Hough Transform has been widely successful in meeting the real time requirement of being able to detect the circles on image.

A commonly faced problem in computer vision is to determine the location, number or orientation of a particular object in an image. One problem could for instance be to determine the straight roads on an aerial photo, this problem can be solved using Hough transform for lines. Often the objects of interest have other shapes than lines, it could be parables, circles or ellipses or any other arbitrary shape. The general Hough transform can be used on any kind of shape, although the complexity of the transformation increase with the number of parameters needed to describing the shape. In the following we will look at the Circular Hough Transform (CHT).

II. CIRCULAR HOUGH TRANSFORM

One of the most commonly used algorithms to recognize different shapes in an image is Hough Transform [13]. Hough Transform was introduced by Paul Hough in 1962 and patented by IBM. In 1972 Richard Duda and Peter Hart modified Hough Transform, which is used universally today under the name Generalized Hough Transform [14]. An extended form of General Hough Transform, Circular Hough Transform (CHT) [13], is used to detect circles. The edge detected from the Canny edge detector forms the input to extract the circle using the Circular Hough Transform.

In Circular Hough Transform, voting procedure is carried out in a parameter space. The local maxima in accumulator space, obtained by voting procedure, are used to compute the Hough Transform. Parameter space is defined by the parametric representation used to describe circles in the picture plane, which is given by equation (1.1). An accumulator is an array used to detect the existence of the circle in the Circular Hough Transform. Dimension of the accumulator is equal to the unknown parameters of the circle. The equation of the circle in parametric form is given by

$$(X - X_0)^2 + (Y - Y_0)^2 = r^2 \qquad \ldots\ldots (1)$$

Equation (1.1) implies that the accumulator space is three-dimensional (for three unknown parameters x_0, y_0 and r). Points corresponding to x_0, y_0 and r, which has more votes, are considered to be a circle with center (x_0, y_0) and radius r.

A Practical Problems With Hough Transform

Quantization errors: An appropriate size of accumulator array is difficult to choose.

Difficulties with noise: The advantage of Hough Transform is that it connects edge points across the image to some form of parametric curve. However, this is also its weakness, because it is possible that good phantom circles might also be detected with a reasonably large voting array.

Time taken: A high dimensional parameter space can slow the Hough Transform process.

Memory: As the accumulator is used in random-accessed fashion, it can easily over run the available memory.

III. PROPOSED METHOD

A. Canny Edge Detection

Edge detectors of some kind, particularly step edge detectors, have been an essential part of many computer vision systems. The edge detection process serves to simplify the analysis of images by drastically reducing the amount of data to be processed, while at the same time preserving useful structural information about object boundaries. The Canny operator works in a multi-stage process. First of all the image is smoothed by Gaussian convolution. Then a simple 2-D first derivative operator (somewhat like the Roberts Cross) is applied to the smoothed image to highlight regions of the image with high first spatial derivatives. Edges give rise to ridges in the gradient magnitude image. The algorithm then tracks along the top of these ridges and sets to zero all pixels that are not actually on the ridge top so as to give a thin line in the output, a process known as non-maximal suppression. The tracking process exhibits hysteresis controlled by two thresholds: T1 and T2, with T1 > T2. Tracking can only begin at a point on a ridge higher than T1. Tracking then continues in both directions out from that point until the height of the ridge falls below T2. This hysteresis helps to ensure that noisy edges are not broken up into multiple edge fragments.

B. Hough Transform

The Hough transform is a mapping from the image plane onto the parameter space. The parameter space is quantized into an accumulator array, and each accumulator stands for the curve specified by the coordinates of the accumulator. For each edge point on the image plane, the curves passing through the point are computed, and the accumulators corresponding to these curves are incremented by 1. After the transform, the accumulator with a peak value indicates the existence of a curve, which is specified by the coordinates of the accumulator, on the image plane.

Choosing (xo, yo, r) as the parameters of the Hough transform, the locus of the accumulators, in the parameter space, incremented by an edge point on the image plane is a right circular cone (see Fig. 2). If there is a circle on the image plane, all the right circular cones incremented by the edge points of the circle will intersect at a common accumulator in the parameter space.

The coordinates of this common accumulator are the parameters of the equation for the circle on the image plane. The Hough transform is capable of detecting partial shapes. Treating a circular arc as a partial shape of a circle, the Hough transform for detecting circles can be applied to detect circular arcs

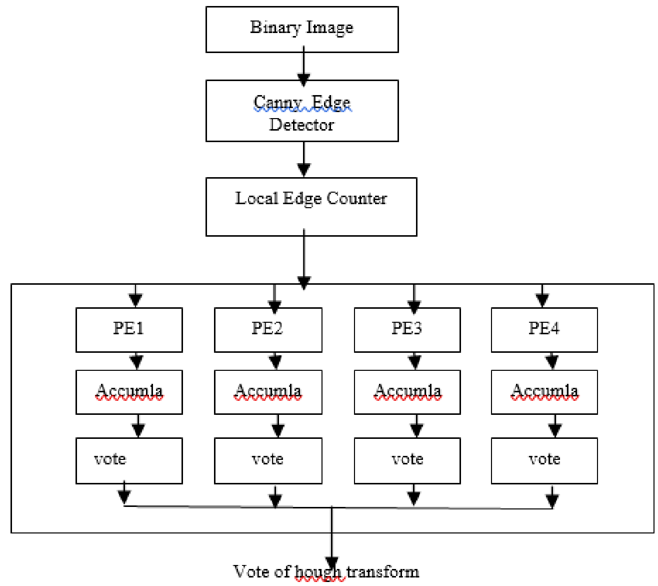

Fig 1: Proposed Architecture

A peak value in the accumulator array indicates the existence of a circular arc. The center and radius of the detected arc are given by the coordinates of the peak.

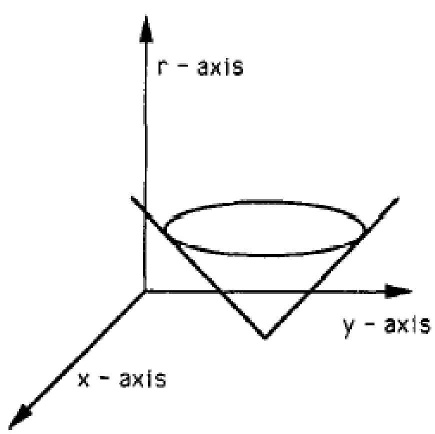

Fig 2:Hough Transform space

C. Accumulator

First we find all edges in the image. This step has nothing to do with Hough Transform and any edge detection technique of your desire can be used. It could be Cannyl operations. At each edge point we draw a circle with center in the point with the desired radius. This circle is drawn in the parameter space, such that our x axis is the a - value and the y axis is the b value while the z axis is the radii. At the coordinates which belong to the perimeter of the drawn circle we increment the value in our accumulator matrix which essentially has the same size as the parameter space. In this way we sweep over every edge point in the input image drawing circles with the desired radii and incrementing the values in our accumulator. When every edge point and every desired radius is used, we can turn our attention to the accumulator. The accumulator will now contain numbers corresponding to the number of circles passing through the individual coordinates. Thus the highest numbers (selected in an intelligent way, in relation to the radius) correspond to the center of the circles in the image.

D. Circle Detection

This is not a part of the Hough Transform it desirable to be able to find circles from the accumulator data. If no a priori knowledge is known about the number of circle and their radii then this process can be quite challenging. One approach is to find the highest peaks for each a,b plane corresponding to a particular radius, in the accumulator data, If the height of the peak(s) is equal compared to the number of edge pixels for a circle with the particular radius, the coordinates of the peak(s) does probably correspond to the center of such a circle. But the center of a circle can also be represented by a peak with a height less than the number of edge pixels, if for instance the circle is not complete or is ellipse shaped. If it is difficult to locate exact peeks, the accumulator data can be smoothed.

IV. RESULT AND COMPARISION

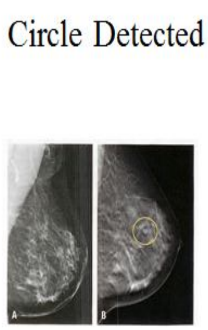

Fig 3:Input Image

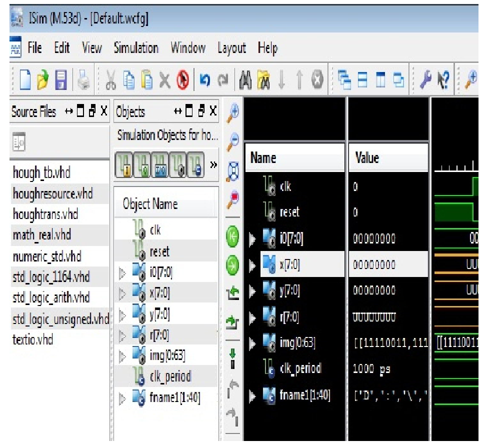

Fig:4 Circle detection output

TABLE I: Image specifications and execution time

Images	Execution time(ms)
Sailboat(line)	2.86(VLSI)
House(line)	2.88(VLSI)
Camera man(line)	2.52(VLSI)
Coin(circle)	23.33(MATLAB)
Art(circle)	23.31(MATLAB)
Test image 1(circle)	2.09(VLSI)
Test image 2(circle)	2.17(VLSI)

Table II Memory Bandwidth Of Direct Implementation Of Hough Transform And The Proposed Architecture

Image	Consumed Memory Bandwidth (bits)	
	Direct Implementation	Proposed Architecture
Airplane	31,862,160	1,172,880
Baboon	42,230,160	1,172,880
Brick	38,455,560	1,172,880
House	34,953,120	1,172,880
Lena	28,100,520	1,172,880
Peppers	23,130,360	1,172,880
Sailboat	35,727,480	1,172,880
San Diego	35,397,000	1,172,880

V. CONCLUSION

The simple approach is proposed in this project is a resource efficient architecture for calculating Hough transform. The incrementing property for both inter block and intra block incrementing are used to reduce the resource requirement. Two accumulators are used to facilitate the inter block incrementing, and zero blocks are skipped by introducing a run length coding scheme. The intra block incrementing could efficiently reduce the resource requirement. Instead of computing the ρ of every pixel in a block, vote offset is more efficient to determine the corresponding votes. The locality of a block is analyzed and the votes corresponding to an identical are consolidated in order to reduce the memory access and fully utilize the FPGA memory bandwidth. The result shows that the proposed PE could achieve the best throughput with the same amount of resources compared to previously reported architectures. The proposed PE is implemented on an Altera kit device and the maximum frequency is 200 MHz.

REFERENCES

[1] J. Canny, “A computational approach to edge detection,” IEEE Trans. Pattern Anal.and Mach. Intell., vol. 8, pp. 679-714, Nov. 1986.

[2] L. S. Davis, "A survey of edge detection techniques," Comput. Graph. and ImageProcess., vol. 4, pp. 248-260, Sept. 1975.

[3] M. Grazia Albanesi, “Time complexity evaluation of algorithm for the houghtransform on mesh-connected computers,” in Proc. 5th Annu. European ComputerConf., Bologna , Italy , 1991, pp. 253-257.

[4] P. Y. Hsiao, C. H. Chen, S. S. Chou, L. T. Li, and S. J. Chen, “A parameterizabledigital-approximated 2D Gaussian smoothing filter for edge detection in noisyimage,” in IEEE Inter. Symp. Circuits and System, Island of Kos, Greece, 2006, pp.3189-3192.

[5] D. Forsyth and J. Ponce, Computer Vision: A Modern Approach. Upper Saddle River,NJ: Prentice Hall, 2003.

[6] R. C. Gonzalez and R. E. Woods, Digital Image Processing Second Edition[M].Upper Saddle River, NJ: Pearson Education Inc., 2002.

[7] G. M. Schuster and A. K. Katsaggelos, “Robust circle detection using a weighted mse estimator,” in Proc. Inter. Conf. Image Processing, Singapore, 2004, pp. 2111-2114.

[8] Z. Li, Y. Liu, J. Yun, and F. Huang, “A new edge detection method based on contrastenhancement,” in Conf. Future BioMedical Inform. Eng., Sanya, China, 2009,pp. 164- 167.

[9] F. Yan, X. Shao, G. Li, Z. Sun, and Z. Yang, “Edge detection of tank level IRimaging based on the auto-adaptive double-threshold canny operator,” Intell. Inform.Technology Applicat. Research Assoc., vol. 3, pp. 366 - 370, Dec. 2008.53

[10] R. O. Duda and P. E. Hart, “Use of the hough transformation to detect lines andcurves in pictures,” Commun. the ACM, vol. 15, pp. 11-15, Jan. 1972.

[11] M. Nixon and A. S. Aguado, Feature Extraction & Image Processing. Burlington,MA: Academic Press, 2008.

[12] E. R. Davis, Machine Vision: Theory, Algorithms, Practicalities. Burlington, MA:Academic Press, 1990.

[13] G. Q. Lu, H. G. Xu; and Y. B. Li, “Line detection based on chain code detection,” inIEEE Int. Conf. Vehicular Safety, Xian, China, 2005, pp. 98-103.

[14] H. Freeman, “Computer processing of line drawing images,” Computing Surveys,vol. 6, no. 1, pp. 57-97, Mar. 1974.

[15] M. Z. Zhang and H. R. Cao, “A new method of circle’s center and radius detection in image processing[C],” in IEEE Int. Conf. Automation and Logistics, Qingdao, China,2008, pp. 2239-2242.