Hardware Implementation of Hough Transform for Circle Detection

Mrs. Sarika K
Assistant Professor
Department of Electronics & Communication Engineering
Karpaga Vinayaka Collge of Engineering and Technology,
Kanchipuram - 603308, Tamil Nadu, India, Phone (0427) 565486
sarikakmenon@gmail.com

Dr. S. Veni
Associate Professor
Department of Electronics & Communication Engineering
Amrita Vishwa Vidyapeetham,
Coimbatore - 641112, Tamil Nadu, India, Phone (0422) 2686000
s_veni@cb.amrita.edu

ABSTRACT

Hough transform is a feature extraction technique used in image analysis, digital image processing and computer vision. To extract features from digital images, it is useful to be able to find arbitrary shapes like straight lines, circles, ellipses etc in the images. In order to achieve this goal, one must be able to detect a group of pixels that are on a required curve. Most of the elements are geometrically circular in shape and detecting circular shapes is an important task in image processing which is a preprocessing step before image analysis. Detection of circles using Hough Transform is most useful technique in the applications like iris detection, satellite imaginary, medical image processing, satellite imaging, Hyper spectral image processing etc., Hardware implementation of the Circular Hough Transform (CHT) is essential for real time applications. In this paper, the Hardware implementation of CHT using CORDIC algorithm is proposed which has not been addressed yet. The CORDIC algorithm reduces computational complexity by replacing exhaustive arithmetic operations with adders and shifters which is evident from the results obtained from this work.

Keywords
Circular Hough Transform (CHT), Hardware Implementation, Sobel Edge Detector, CORDIC Algorithm.

1. INTRODUCTION

Hough Transform [1] is a popular technique for detecting curves in images and Circular Hough Transform (CHT) particularly is for detecting circles in the absence of proper information. In actual applications, through preprocessing operations like edge detection and thresholding, the images are converted into binary feature images. The pixels with the pixel value "1" are called feature points. A circle is the one that passes through many features points. By detecting the presence of groups of collinear or almost collinear figure point will give the structure of the curve. Actually detecting collinear points mathematically is equivalent problem of finding concurrent lines. This method involves transforming each of the feature points into a curve in a parameter space. For circles the parameter space is 3 dimensional in nature, represented by co-ordinates of centre (a, b) and the radius of the circle (r).

The parametric equation of circle is given by,

$$a = x + r * \cos\theta \qquad (1)$$
$$b = y + r * \sin\theta \qquad (2)$$

Where angle 'θ' varies from 0^0 to 360^0 and range of radius will depend on the size of the image.

For each instant, a circle is drawn for a feature point (x, y), it will produce (a, b) value for the corresponding radius r. It can be considered as a "vote" for the specific (a, b, r) value in the parameter space or the accumulator space. After processing all of the feature points, the value that has the maximum accumulated votes will correspond to the circle that passes through the largest number of feature points. In the implementation, the votes for a specific (a, b, r) value can be stored in a memory addressed by this specific value.

Algorithm – voting procedure of Accumulator space

```
1. Initialize accumulator space to zero
2. for each edge point (x, y)
        for radius 1 < r < (image size/2)
                for angle 0⁰ < θ < 360⁰
                        a = x + r * cos(θ)
                        b = y + r * sin(θ)
                        vote (a, b, r) = vote (a, b, r)+1
                end
        end
end
3. Find one or several maxima in the accumulator
```


ICONIAAC '14, October 10 - 11 2014, Amritapuri, India

The remainder of this paper is organized as follows. Section II provides basic work flow of CHT. Section III describes the hardware implementation of edge detector, CORDIC and proposed CHT module. Validation of results is addressed in section IV and finally conclusion is given in section V.

2. BASICS OF CIRCULAR HOUGH TRANSFORM

2.1 Preprocessing

We are assuming that input image might be noisy, mostly contaminated with salt and pepper. This noise is to be eliminated in order to construct a better edge-map for the next stage. To prevent this salt and pepper Noise, 3 × 3 Median filter is moved along the image for blurring it. Since noise is tremendously destructive for edge detection algorithms, it is substantial to perform this process prior to the next stage. The result is now ready for the edge detection stage.

2.2 Edge Detection

Edge detection is a method of determining the discontinuities in gray level images. Extracting the edges from an image simplifies the analysis of the images by dramatically reducing the amount of data to be processed, while at the same time preserving useful information about the boundaries. In the conventional edge detection mechanism examine image pixels for abrupt changes by comparing pixels with their neighbors. The feature points [2] are found out by calculating gradient value. The magnitude of gradient G will determine the feature points where in most of them are zero values for G and it is possible to eliminate them by thresholding technique. Then that matrix contains most of the picture edges of interest.

The sobel operator [3] is a classic first order edge detection operator. The sobel operator only considers the two orientations which are 0°and 90°convolution kernels, i.e. horizontal mask G_x and vertical mask G_y as shown in Figure 1.

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Figure 1. Horizontal and vertical masks of Sobel Edge Detector [3]

2.3 Circular Hough Transform

In this stage we intend to estimate the circles which are visible on the edge-map. The equation of the circle is represented as,

$$(x-a)^2+(y-b)^2=r^2 \tag{3}$$

We know that,

$$\sin^2\theta+\cos^2\theta=1 \tag{4}$$

Equation (3) can be rewritten as,

$$(x-a)^2+(y-b)^2=r^2*(\sin^2\theta+\cos^2\theta) \tag{5}$$

The cosine function represents X-axis and sine function represents Y-axis. Therefore after equating we get,

$$(x-a)^2=r^2*\cos^2\theta \tag{6}$$

$$(y-b)^2=r^2*\sin^2\theta \tag{7}$$

Take square root on both sides of the equations (6) and (7),

$$(x-a)=r*\cos\theta \tag{8}$$

$$(y-b)=r*\sin\theta \tag{9}$$

From above equations we get the parametric equation of circle as (1) and (2).

Circle has 3 parameters, co-ordinates of centre (a, b) and radius 'r'. So the parameter space for CHT is a 3-D matrix with x and y axis being pixels along width and height of the image and z axis demonstrating different radii. It shows parameter space for a circle which belongs to R^3 and the line which belongs to R^2. Hence, the complexity of Hough Transform also increases.

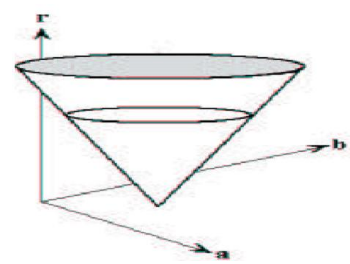

Figure 2. Three dimensional Accumulator space of CHT [7]

In Circular Hough Transform, for each edge point on the edge map we have to draw a circle with center in the edge point with radius r and increment all coordinates that the perimeter of the circle passes through the accumulator space [4,5, 6]. When every edge point and every desired radius is used, now concentrate to the accumulator space. Find one or several maxima in the accumulator and map the found parameters (a, b, r) corresponding to the maxima of accumulator back to the original image. For specific applications [7], if the radius is known then the accumulator dimension reduces to 2-D space. Figure. 3 shows the mapping of CHT from (x, y) space to the parameter space when the radius is constant.

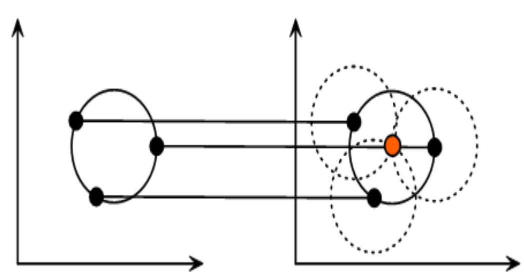

Figure 3. Circular HT from the x,y-space (left) to the parameter space (right) [7]

Flowchart for the CHT algorithm is given in Figure. 4 for the algorithm described in the introduction section.

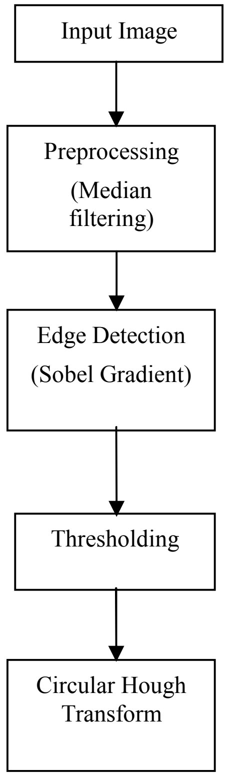

Figure 4. Flowchart of CHT

3. HARDWARE IMPLEMENTATION

3.1 Sobel Edge Detector Architecture

The proposed architecture of sobel edge detector is shown in in Figure. 5. It has 4 main blocks namely 3 x 3 window generations, sobel gradient calculation, control unit and thresholding unit. Initially, data input is given to the 3 x 3 window generation unit which will help to make it appropriate for the sobel mask. The pixel values under consideration are stored in RAM without bothering the neighborhood. As the image is represented by matrix, proper neighborhood selection is required for 2D convolution. Let the size of given image will be n x n. The order of the pixel value address is 0, n and 2n respectively for the first consecutive three clock cycles. For the second three consecutive clock cycles the address of the pixel values are to be taken as 1, n+1 and 2n+1 and so on.

Second unit (sobel gradient calculation) is used to calculate magnitude of the first order gradient by using the equation

$$|G| = |G_x| + |G_y| \qquad (10)$$

Where G_x is horizontal sobel mask and G_y is the vertical sobel mask. In thresholding unit, edge pixels which are the image edges of interest are computed by choosing proper threshold for the sobel output G. Control unit is designed to control the whole architecture by providing proper clock, reset and enable.

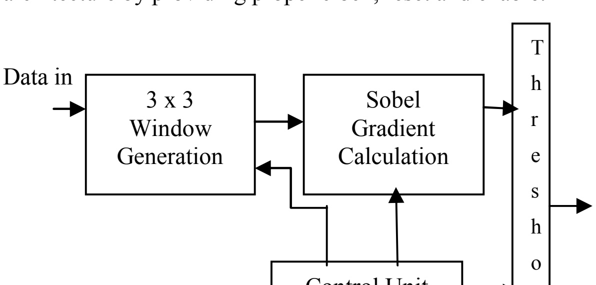

Figure 5. Proposed architecture for Sobel Edge Detector

3.2 CORDIC Architecture

To generate the trigonometric angles Cordic Algorithm is used which allows trigonometric angles required for CHT and can be calculated primarily by shifting and adding. The CORDIC algorithm first proposed by Volder [8], this method is very effective because it avoids the multiplication terms. The parameter estimation of circle requires exhaustive arithmetic operations like multiplication, square root evaluation, division, addition/subtraction and squaring [9,10]. To reduce the computation and hardware requirements for the estimation of these parameters, the problems are reformulated in terms of the CORDIC rotation by parametric equation of circle.

An initial vector $[x\ y]^T$ undergoing a rotation through an angle ϕ, will generate the final vector $[x'\ y']^T$ according to the following relation.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \qquad (11)$$

By simplifying above matrix,

$$x' = x\cos\phi - y\sin\phi \qquad (12)$$

$$y' = y\cos\phi + x\sin\phi \qquad (13)$$

We can rearrange this as,

$$x' = \cos\phi\,[x - y\tan\phi] \qquad (14)$$

$$y' = \cos\phi\,[y + x\tan\phi] \qquad (15)$$

The total rotation ϕ can be expressed in the steps of smaller angles ϕ_i s, in such a way that,

$$\phi = \sum_{i=1}^{N} \phi_i \qquad (16)$$

where N is an integer.

From equation (16) by cascading a number of elementary rotational stages as follows,

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \prod_{i=1}^{N} \begin{bmatrix} \cos\phi_i & \sin\phi_i \\ -\sin\phi_i & \cos\phi_i \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \qquad (17)$$

We know if the angle is too small sine and tangent of angle will be approximated to the given angle. Here if we take elementary angle is small enough such that, $\sin\phi_i = \phi_i = 2^{-i}$ and $\cos\ \phi_i = 1 - 2^{-(2i+1)}$

So equation (17) can rewritten as,

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \prod_{i=1}^{N} \begin{bmatrix} 1-2^{-(2i+1)} & 2^{-i} \\ -2^{-i} & 1-2^{-(2i+1)} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (18)$$

On every iteration, a decision is made whether to add or subtract the next value of ϕ. This is made by comparing ϕ with the given angle. The values for ϕ are chosen such that tan (ϕ) is a fractional power of 2, the values are given in table 1.

Table 1. values ϕ and corresponding tangent values

Tangent of ϕ	ϕ
Tan (ϕ) = 1/1	ϕ_1 = 45
Tan (ϕ_2) =1/2	ϕ_2 = 26.5650
Tan (ϕ_3) = 1/4	ϕ_3 = 14.0362
Tan (ϕ_4) = 1/8	ϕ_4 = 7.12502
Tan (ϕ_5) = 1/16	ϕ_5 = 3.57633
Tan (ϕ_6) = 1/32	ϕ_6 = 1.78991
Tan (ϕ_7) = 1/64	ϕ_7 = 0.89517
Tan (ϕ_8) = 1/128	ϕ_8= 0.447614
Tan (ϕ_9) = 1/256	ϕ_9 = 0.2238
Tan (ϕ_{10}) = 1/512	ϕ_{10} = 0.1119
Tan (ϕ_{11}) = 1/1024	ϕ_{11} = 0.055953

Figure. 6 shows the architecture of CORDIC [10] unit to calculate trigonometric functions sine and cosine. On the positive edge of the enable called 'START', system is initiated to work. The proper value of sine and cosine will get only if the enable 'DONE' is high.

The circuit diagram consists of mainly two parts, one is for angle up gradation and other is for sine and cosine calculation. ATAN LUT consists of the values of angles that are used during the calculation. The range of ϕ depends on the accuracy of the given system. The register Z REG store values of angle ϕ. At first stage Z REG initialized to zero. After that on each iteration, Z REG updates with the new value. The control bit of multiplexer Z_{msb} will determine whether we have to add the angle on ATAN LUT with Z REG or subtract. The process continues till counter reaches the value 'N'. Final value of Z REG will be the target angle.

Second part of CORDIC circuit is designed for calculation of sine and cosine values. Registers X REG and Y REG initialized to 1 and 0 respectively. If we initialize X REG to 1 finally we have to multiply the final result with the aggregate constant K. To avoid this initialize X REG as K (= 0.607253). At each clock cycle the value for X and Y is calculated according to the corresponding angle in the ATAN LUT i.e less than or greater than that of given angle. After N iterations we get proper values of cosine and sine from X REG and Y REG respectively.

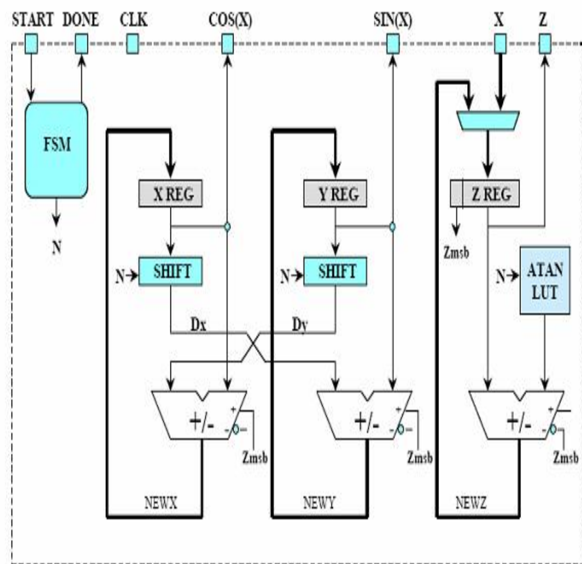

Figure 6. Architecture for CORDIC [10]

3.3 Circular Hough Transform Architecture

The co-ordinates of edge detected values are stored in 2 RAMs X and Y for x co - ordinates and y co-ordinates respectively [12]. CHT hardware proposed by Tsung Hanet Tsai et.al. [13] has CORDIC unit that computes trigonometric functions sine and cosine which are needed for the calculations of parametric equation of circle. By equation (2) and (3), (x, y) is the value of the co-ordinates with sobel output whose value is "1". This means the co-ordinates of the feature points. 'r' the radius of the circle ranges up to half of the size of the image we give. For some specific applications like industrial applications value of 'r' may take as constant. The angle 'θ' varies from 0^0 to 360^0 with a small step size k.

The proposed architecture of Circular Hough Transform is shown in Figure 7. The CORDIC unit calculates the value of sinθ and cosθ by an accuracy of 4 bit after the decimal point. The angle 'θ' increments in each clock cycle. The clock designed in such a way that CORDIC unit will give proper output for sinθ and cosθ, in another words when enable 'DONE' signal is high. The raw register X contains the x index value and raw register Y contains the y index value of sobel output whose bit value is '1'. For the calculation of 'a' and 'b' using equation (1) and (2), each values of x and y added with r*cosθ and r*sinθ where 'r' is the radius. For each 'r',' θ' varies from 0^0 to 360^0 and for each (x, y) pair 'r' varies from a small value to the half of the image size.

Control Unit controls the total circuit to work properly. CORDIC unit takes 9 clock cycles to determine sine and cosine value with an accuracy of 4 decimal points. After each 18 clock cycles value of 'θ' increments by 5^0. That means total of 73 times 'θ' increments for one edge point (x, y). For this 73 x 9 clock cycles are required. Each elements of the image matrix under goes this calculation to determine (a, b) pair.

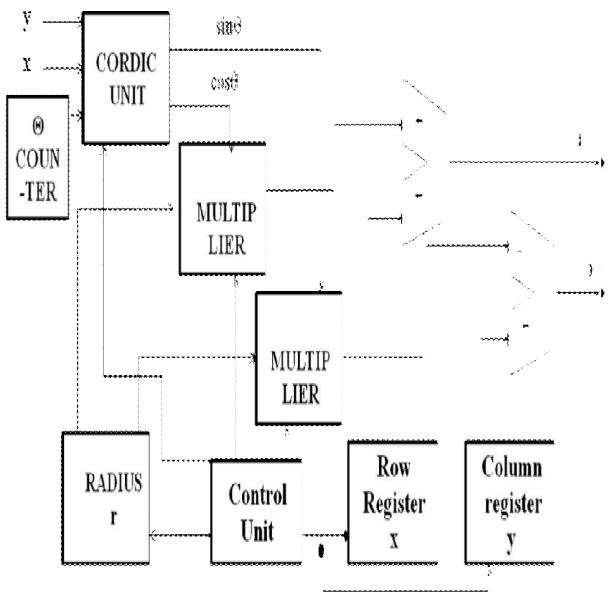

Figure 7. Proposed hardware architecture for CHT

3.4 Accumulator

Accumulator of CHT is 3 dimensional, because circle has three parameters, co-ordinates of circle and radius (Refer Fig.2). After calculating values of (a, b) we have to increment the value of 3D-accumulator whose index value is (a, b, r). The size of accumulator for each radius will be equal to the size of the image. For each value of x, y and r, the parameters and b are calculated by using proposed hardware architecture of CHT, i.e., for each edge point in circle (x, y), increment all points in the 3-D accumulator array (a, b, r). One or more local maxima in the accumulator array correspond to centers of circles in the image can be detected using this method. Address value of accumulator will be concatenated value of ‘a’ and ‘b’, where ‘a’ points the location and shows the drift. After finding out the address, value of that location is incremented by one.

Figure. 8 shows the proposed hardware implementation for calculation of parameters of circle. RAM (Accumulator) contains values of (a, b) of corresponding radius ‘r’. During Reset, the register is initialized to zero. During the first clock cycle the value of the accumulator with address 0000H is given as input to the comparator. This value of the accumulator is compared with the value on the register and greater value is stored by replacing the previous register value. Comparator is basically a full subtractor and the accumulator value is subtracted from value in the register. If borrow of subtractor is ‘0’, address value greater than that of value on the register replaces register value by accumulator value. Else borrow is ‘1’ implies register contains higher value and no replacement have to be done. In this way at each clock cycle values are compared and finally maximum value of accumulator can be determined. The address of the maximum value of accumulator gives the co-ordinates of centre of the circle {a, b} and corresponding ‘r’ will be its radius.

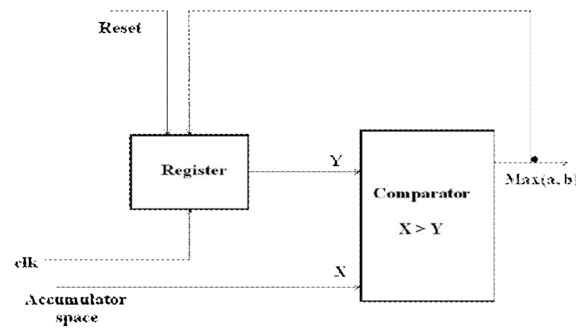

Figure 8. Proposed Architecture for Calculating Parameters (a, b, r)

4. RESULTS AND DISCUSSIONS

4.1 Matlab Implementation Results

Median filter, Sobel Edge Detector, CORDIC and CHT modules are simulated in iverilog and output was verified with MATLAB software. The input for the proposed hardware module is given as text file from MATLAB. After processing the derived output datas from the hardware has been taken as text file and the images were displayed using MATLAB.

The expected output of hardware implementation results of CHT are the parameter values of circle radius (r) and co - ordinates of the centre (a, b). These values are used to draw the circle using Matlab software. The resulted output for the different images of CHT of Matlab implementation and Hardware implementation are shown in Figures 9 - 12.

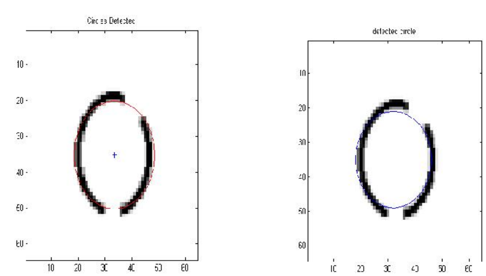

Figure 9. (a) Matlab Output and (b) Hardware implementation output of Circular Hough Transform of image circle.png

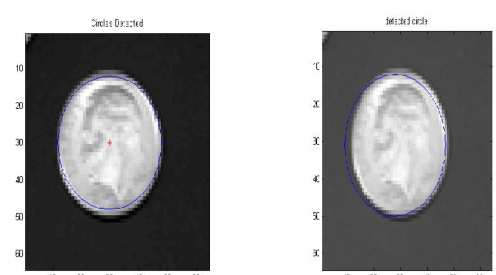

Figure 10. (a) Matlab Output and (b) Hardware implementation output of Circular Hough Transform of image coin.png

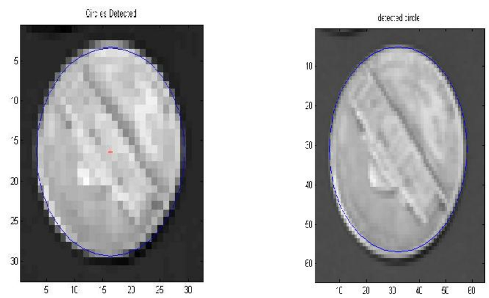

Figure 11. (a) Matlab Output and (b) Hardware implementation output of Circular Hough Transform of image coin1.png

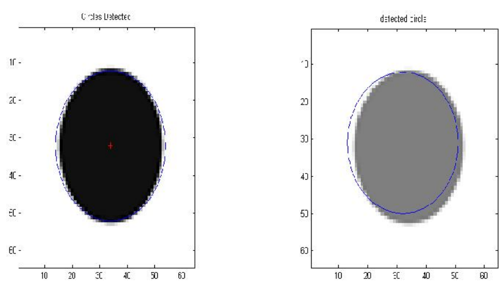

Figure 12. (a) Matlab Output and (b) Hardware implementation output of Circular Hough Transform of image disk.png

4.2 Hardware Implementation Results

Hardware implementation of Circular Hough Transform was performed for set of image with circular objects. The performance of the proposed hardware was evaluated in terms of accuracy and determination of parameters of circle co-ordinates (centre and radius) which are shown in Table.2.

Table 2. Parameters of circle from Matlab and Hardware output for different Images

Images	Computed Parameters (a, b, r) using Matlab	Computed Parameters (a, b, r) using Hardware
Circle.png	(33, 35, 15)	(32, 35, 14)
Coin.png	(29, 30, 18)	(28, 31, 19)
Coin1.png	(32, 32, 26)	(32, 31, 26)
disk.png	(34, 32, 20)	(32, 31, 19)

Hardware implementation of CHT was performed successfully which is evident from the above results. It is possible to obtain the similar results compared with the Matlab implementation i.e.,. The parameter values (a, b, r) computed using hardware implementation were used to detect circles in the images and these images were displayed using Matlab software.

4.3 Synthesis Result

Synthesis of architecture was performed in Quartus II software targeting Cyclone II EP2C20F484C7 FPGA. The RTL view of CHT module (for the architecture shown in Figure. 7) is given in Figure. 13. Hardware utilization details of the proposed architecture modules are given in Table. 3.

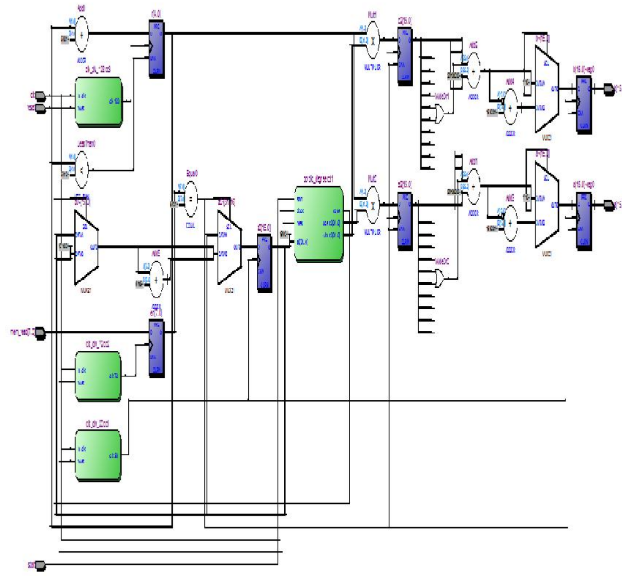

Figure 13. RTL view of Circular Hough Transform hardware module

Table 3. Accumulator Module (Quartus Synthesis Result in Cyclone II EP2C20F484C7 FPGA)

Characteristics	Median Filter	Sobel Edge Detector	Circular Hough Transform
Total Logic Elements	942	387	3204
Total Combinational Functions	942	345	2133
Total registers	137	223	1286
Total Thermal Power Dissipation	68.69mW	71.27mW	76.52mW
Maximum Frequency	380.1MHz	100.6MHz	75.8MHz

5. CONCLUSIONS

The Hardware implementation of Circular Hough Transform for circle detection has been proposed and implemented successfully. In the preprocessing stage, noise contaminated with image was eliminated by using Median Filter in order to construct a better edge map. Since Hough Transform is a edge linking process edge detection was performed as a preprocessing stage. The proposed Sobel Edge detection architecture was used for calculating edges. The proposed architecture of Circular Hough Transform was designed using CORDIC algorithm. It reduces the computational complexity by replacing exhaustive arithmetic operations with adders and shifters.

Due to the limitation in the memory it was possible to synthesis 64 x 64 image for hardware implementation. In the future work the architecture can be extended for large size images with the advanced FPGA's and the work can be extended for real time applications using Xilinx System Generator with suitable hardware platforms. The performance of Circular Hough Transform can still be improved with optimized architecture.

6. REFERENCES

[1] E. O. Duda and P. E. Hart, "Use of the Hough transform to detect lines and curves in pictures," Commun. ACM, pp. 11-15, 1972.

[2] Frank O'gorman and M. B. Clowes, "Finding Picture Edges Through Collinearity of Feature Points", IEEE Trans. Comput., vol. C-100, pp. 449-456, 1976.

[3] R C Gonzalez and R E Woods, "Digital Image Processing", 2nd ed. Prentice Hall, 2002.

[4] Ms.Priyanka.S.Chikkali and prof. K.Prabhushetty, "FPGA based Image Edge Detection and Segmentation", AES&T, pp. 187-192, 2011.

[5] Nawal Alioua, Aouatif Amine, Mohammed Rziza and Driss Aboutajdine, "Eye state analysis using iris detection based on Circular Hough Transform", IEEE 2010.

[6] Xing Chen, Ling Lu and Yang Gao, "A New Concentric Circle Detection Method Based on Hough Transform", ICCSE, pp. 753 - 758, 2012.

[7] Sirisak Liangwongsan, Boonraung Marungsri, Ratchadaporn Oonsivilai, Anant Oonsivilai, "Extracted Circle Hough Transform and Circle Defect Detection Algorithm", World Academy of Science, Engineering and Technology, pp. 432-437, 2011.

[8] J.E. Volder, "The CORDIC trigonometric computing technique", IRE Trans. Electron. Computer. EC 8, pp. 330-334, 1959.

[9] K. Maharatna and S. Banerjee, "A VLSI array architecture for Hough Transform," Pattern Recognition, vol. 34, no. 7, pp. 1503-1512, 2001.

[10] Pramod K Meher, Javier Valls, Tso-Bing Juang, K. Sridharan and Koushik Maharatna, "50 years of CORDIC: Algorithms, Architecture and Applications", IEEE transaction on Circuits and Systems, vol. 56, pp. 1893 – 1907, 2009.

[11] U Meyer-Baese, "Digital Signal Processing with Field programmable Gate Arrays", 3rd ed. Springer, 2001.

[12] Jau Ruen Jen, Mon Chau Shie and Charlie Chen, "A Circular Hough Transform Hardware for Industrial Circle Detection Applications", ICIEA 2006.

[13] Tsung Han Tsai, Chia-Hao Yeh and Yu-Jung Huang, "Cordic Architecture for Hough Transform Applications", IEEE, pp. 442 – 445, 2008.