A COARSE TO FINE SEARCH TECHNIQUE TO DETECT CIRCLES IN IMAGES

M. Atiquzzaman
Dept. of Computer Science and Computer Engineering
La Trobe University, Melbourne 3083, Australia.
Tel: (03) 479 1118, Fax (03) 470 4915
atiq@LATCS1.lat.oz.au

ABSTRACT

Hough transform can be used to detect parametric patterns, such as straight lines and circles, embedded in noisy images. The large amount of storage and computing power required by the Hough transform presents a problem in real-time applications. Multiresolution Hough transform (MHT) very efficient in reducing the computing and storage requirements. In this paper, we show the effectiveness of the Multiresolution Hough transform in detecting circles in images.

1 INTRODUCTION

Patterns in real-world images are frequently found to be discontinuous and embedded in noise. The Hough transform [1] has long been known to be an efficient technique to detect discontinuous patterns embedded in real-world noisy images. For detecting circles, the transform votes for all the possible circles passing through the point. The circle receiving the maximum vote is found by finding the peak in the accumulator array. A comparative study of detecting circles in images using the HT has been discussed in [2].

The transform has the *drawbacks* of being highly compute bound and requiring a large amount of storage. The amount of computation increases with an increase in the size of the accumulator array and the accuracy with which the parameters are to be determined. The resolution of the accumulator array determines the accuracy with which the parameters can be determined.

A considerable amount of research has been devoted to increasing the computational and storage efficiency of the HT. The use of gradient information is known to reduce the computing time by one-sixth when compared to the method without using the gradient information. *Multiprocessor* implementations of the transform have been proposed to reduce the execution time [3, 4].

Coarse-to-fine search strategies [5, 6] are computationally efficient algorithms which are suitable for implementation in single processor systems. The reduced computational complexity of the MHT results from the use of a simple peak detection algorithm in addition to multiresolution images and accumulator arrays in the successive iterations of the algorithm. The effectiveness of the MHT was demonstrated by applying it to images containing straight lines [5]. Since the Hough transform does not provide the length and the end points of a straight line, two algorithms for the detection of the length and end points were proposed in [7, 8]. Among the HT algorithms based on a dynamic quantization, the MHT is still the most efficient algorithm among the ones available in the literature [9].

The *objectives* of this paper are to demonstrate the effectiveness of the MHT in detecting circles, and determine the accuracy with which the parameters of the circle can be detected. The effectiveness will be measured by the rate of convergence of the estimated parameters towards the actual values of the parameters as a function of the number of iterations.

The MHT for detecting circles is described in Section 2 followed by results and conclusions in Sections 3 and 4 respectively.

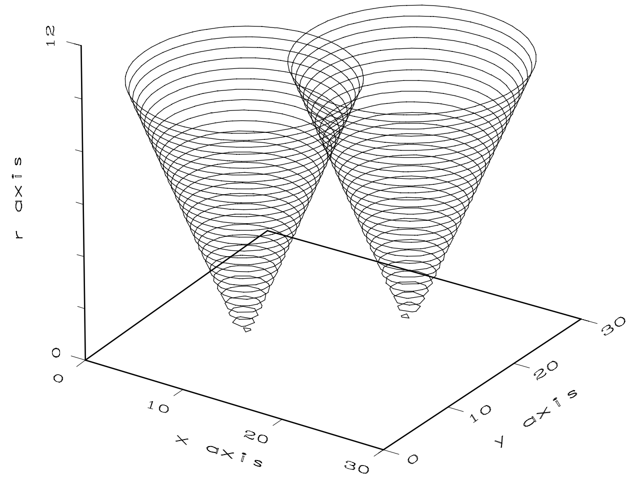

Figure 1: Illustration of the 3D parameter space resulting from two points, (10,10) and (20,20), of a circle having a radius of 5 and the center at (15,15).

2 MHT FOR CIRCLE DETECTION

Let us define a circle in a binary edge image by

$$(x-a)^2 + (y-b)^2 = r^2$$

where (a, b) are the coordinates of the center and r is the radius of the circle. The standard Hough transform [1] requires a 3-dimensional accumulator array having the ranges of a, b and r as $0 - \mathcal{S}(x)$, $0 - \mathcal{S}(y)$ and $0 - \frac{1}{2}\sqrt{(\mathcal{S}(x))^2 + (\mathcal{S}(y))^2}$ respectively where $\mathcal{S}(x)$

and $\mathcal{S}(y)$ are the sizes of the binary image in the x and y directions. The detection accuracy depends on the resolution of the accumulator array, resulting in the accumulator arrays being exorbitantly large if the parameters are to be detected with a reasonable accuracy.

The MHT uses L iterations, using a different image from a set of L images at each iteration. The set of images are generated by reducing the original image $(L-1)$ times. The factor by which an image is reduced at each step is σ. The first iteration uses the smallest image from the set and accumulates the votes in a small accumulator array. The second iteration uses a larger image and a larger accumulator array than those used in the first iteration. However, the parameter range of investigation is narrowed down during the second iteration. The estimates of a, b and r obtained in the first iteration are used to select a reduced range of the parameters to be investigated in the second iteration. The above procedure of reducing the parameter ranges and using increasingly larger images and accumulator arrays at successive iterations of the MHT is carried out until the original image has been analyzed. The above coarse-to-fine analysis technique results in a reduced amount of computation as compared to the original HT using a single image and a single accumulation of the HT. We introduce below some notations which will be used to illustrate the MHT for the detection of circles.

2.1 NOTATIONS

L = Number of iterations in the MHT.

$\mathcal{S}(x,i)$, $\mathcal{S}(y,i)$ = The x and y sizes of the image after i reductions in size. $\mathcal{S}(x,0)$ and $\mathcal{S}(y,0)$ are therefore, the sizes of the original image. We will assume square images whose sizes will be represented by $\mathcal{S}(f,i) = \mathcal{S}(x,i) = \mathcal{S}(y,i) = \mathcal{S}(r,i)$ for $0 \leq i \leq L-1$.

$\mathbf{F} = \{\ f^i_{x,y},\ 0 \leq i \leq L-1,\ 0 \leq x \leq \mathcal{S}(x,i),\ 0 \leq y \leq \mathcal{S}(y,i)\ \}$ = The set of images used at the different iterations in the MHT. Note that $f^0_{x,y}$ is the original image and $f^i_{x,y}$ is used during the $(L-i)$-th iteration.

$\mathcal{S}(a,i)$, $\mathcal{S}(b,i)$, $\mathcal{S}(r,i)$ = The dimensions of the accumulator array along the a, b and r axes during the $(L-1)$-th iteration of the MHT.

$\mathcal{S}(f,i)$ = The size of the image along each dimension at the $(L-i)$-th iteration, assuming square images.

V $= \{ \, v^i_{a,b,r}, \; 0 \le i \le L-1, \; 0 \le a \le \mathcal{S}(a,i), \; 0 \le b \le \mathcal{S}(b,i), \; 0 \le r \le \mathcal{S}(r,i) \, \}$ = The set of accumulator arrays used to accumulate the votes at the different iterations of the MHT.

$\Delta(a, L-i)$, $\Delta(b, L-i)$, $\Delta(r, L-i)$ = Discretization steps of a, b and r during the i-th iteration.

$\mathcal{R}(a, L-i)$ = Range of a during the i-th iteration.

$\mathcal{R}(b, L-i)$ = Range of b during the i-th iteration.

$\mathcal{R}(r, L-i)$ = Range of r during the i-th iteration.

2.2 REDUCTION OF PARAMETER RANGES

The MHT [5] uses small image and accumulator arrays during the initial iterations. If the parameter range is reduced too much (based on the rough estimates) during the initial iterations, it has been found that the actual parameters frequently fall outside the new reduced range of investigation. We overcome the problem by reducing the parameter ranges logarithmically (instead of linearly) at the successive iterations.

Let the a, b and r ranges be reduced by γ after the L-th iteration. To apply a logarithmic range reduction, the parameter range should be reduced by γ/σ^{L-i} after the i-th, $1 \le i \le L$, iteration. If the a, b and r parameters are to be determined with an accuracy of one pixel after L iterations, γ for any parameter can be obtained from the following equation

$$\frac{1}{\gamma/\sigma^0}\frac{1}{\gamma/\sigma^1}\frac{1}{\gamma/\sigma^2}\cdots\frac{1}{\gamma/\sigma^{L-1}} = \frac{1}{\mathcal{S}(f,0)}$$

$$\text{or,} \qquad \gamma = \sqrt[\frac{1}{L}]{\sigma^{\frac{L}{2}(L-1)}\mathcal{S}(f,0)} \tag{1}$$

Without loss of generality, we will assume that the sizes of an accumulator array along the a, b and r dimensions are the same, and the ranges for a, b and r are also reduced by the same factor after an iteration of the MHT. Therefore,

$$\mathcal{R}(a, L-1) = \mathcal{R}(b, L-1) = \mathcal{R}(r, L-1) = \mathcal{S}(f, L-1) \tag{2}$$

After the first iteration, the range of a (b and r) is reduced by γ/σ^{L-1}. Therefore,

$$\mathcal{R}(a, L-2) = \frac{\mathcal{R}(a, L-1)}{\gamma/\sigma^{L-1}}\,\sigma = \frac{\mathcal{S}(f, L-1)}{\gamma/\sigma^{L-1}}\,\sigma \tag{3}$$

Similarly it can be shown that

$$\mathcal{R}(r, L-i) = \mathcal{S}(f, L-1)\sigma^{L-1}\sigma^{L-2}\cdots\sigma^{L-1+1}\left(\frac{\sigma}{\gamma}\right)^{i-1} \tag{4}$$

An an example, using an original image of size 512×512, $L = 4$, and $\sigma = 2$, γ comes out to be 13.454 and $\mathcal{R}(a, 3) = 64$, $\mathcal{R}(a, 2) = 76.109$, $\mathcal{R}(a, 1) = 45.255$, and $\mathcal{R}(a, 0) = 13.454$. Therefore, the parameter ranges are reduced by 1.68, 3.36, 6.725 and 13.45 after the first, second, third and fourth iterations respectively. For square accumulator arrays,

$$\mathcal{S}(a, L-i) = \mathcal{S}(b, L-i) = \mathcal{S}(r, L-i) = \frac{\mathcal{S}(f, L-1)}{\sigma^{\mu}}\sigma^{i-1}$$

Therefore,

$$\begin{aligned}\Delta(a, L-i) &= \frac{\mathcal{R}(a, L-i)}{\mathcal{S}(a, L-i)} \\ &= \frac{\sigma^{L-1}\sigma^{L-2}\cdots\sigma^{L-1}}{\gamma^{i-1}}\sigma^{\mu} \qquad (5)\end{aligned}$$

For $\mu = 1$, $\sigma = 2$, $L = 4$ and $\mathcal{S}(f, L-1) = 512$ we have

$$\begin{aligned}\Delta(a,3) = \Delta(b,3) = \Delta(r,3) &= 2 \\ \Delta(a,2) = \Delta(b,2) = \Delta(r,2) &= 1.189 \\ \Delta(a,1) = \Delta(b,1) = \Delta(r,1) &= 0.3535 \\ \Delta(a,0) = \Delta(b,0) = \Delta(r,0) &= 0.05255\end{aligned}$$

3 RESULTS

The detection accuracy of the MHT will be represented by the percentage error between the determined and the actual values of parameters. The percentage errors of a, b and r will be represented by ε_a, ε_b and ε_c respectively. Since the MHT is a coarse to fine iterative process, we will also show the rate at which the search interval for the parameters reduce at successive iterations.

Figure 2 shows the binary multiresolution images used at the different iterations. The initial image is of size 512×512 and having two discontinuities. $L = 4$, $\sigma = 2$, and $\mu = 2$ have been used. The first iteration uses the 64×64 image; the last uses 512×512. Because of using $\sigma = 2$ and $\mu = 2$, the sizes of the accumulator arrays were 16×16, 32×32, 64×64, and 128×128 during iterations 1, 2, 3, and 4 respectively. The constant-r planes of the accumulator arrays (at the

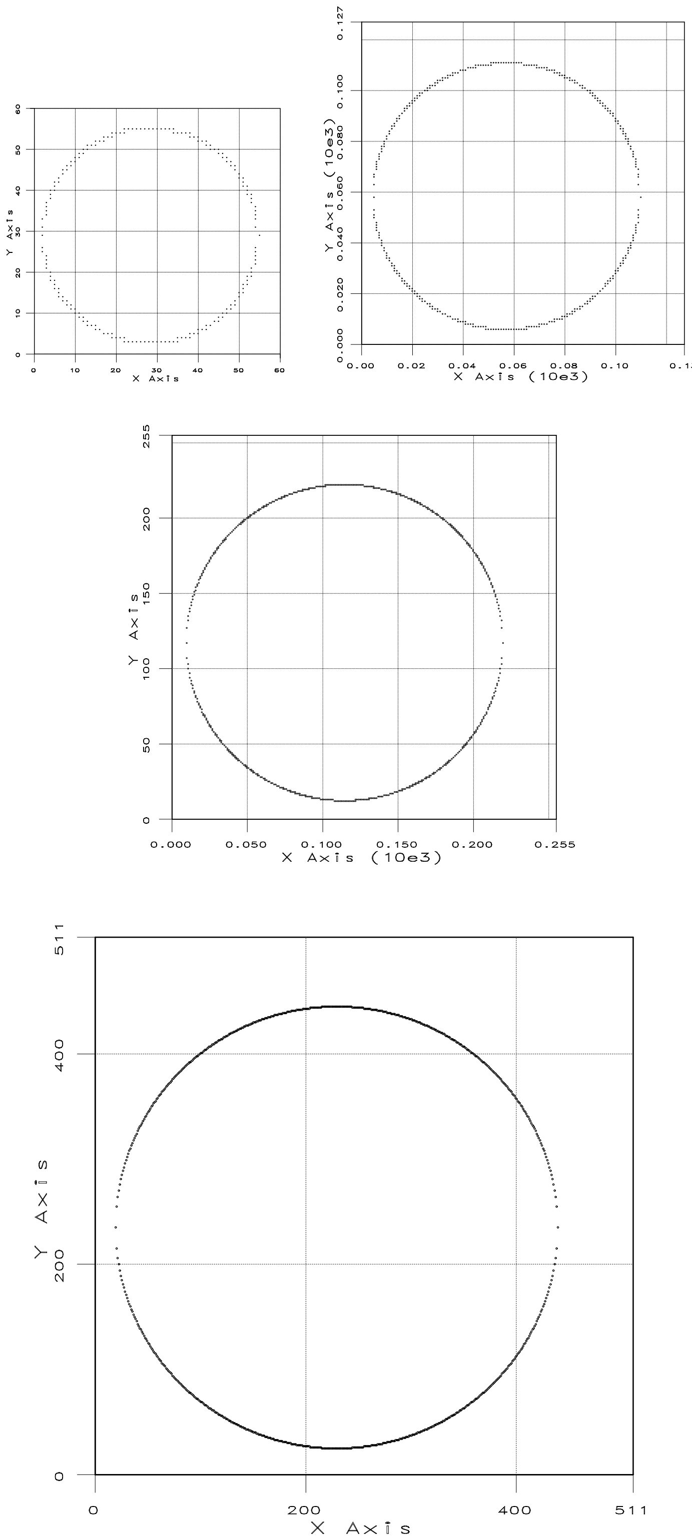

Figure 2: Images used during the different iterations.

Table 1: Percentage errors of a, b and r at each iteration based on 30 images. $\mathcal{S}(f, 0)$=512, $\sigma = 2$ and $L = 4$.

Iter. No.	ε_a		ε_b		ε_c	
	$\mu = 1$	$\mu = 2$	$\mu = 1$	$\mu = 2$	$\mu = 1$	$\mu = 2$
1	8.52	39.02	15.92	15.92	12.66	42.75
2	2.74	4.13	0.83	3.42	3.87	3.87
3	0.17	0.25	0.17	0.27	0.69	0.85
4	0.15	0.15	0.14	0.14	0.17	0.19

different iterations), containing the maximum votes are shown in Figure 3. To show the sharpness of the peak, the planes are represented in three-dimensional forms. Since the first iteration uses the full ranges of the parameter, the position of the peak in the plane during that iteration depends on the actual values of the parameters a and b. As expected, the peak is closer to the center during successive iterations as seen in Figure 3. The peaks are also seen to be very *sharp* and *unique* in the different iterations.

Table 1 presents the percentage errors for $\mu = 1\&2$. The sharp decrease in the percentage errors in successive iterations shows the speed of convergence of the estimated parameters to the actual parameters during the different iterations. Note that the memory required to store the accumulator arrays with $\mu = 2$ is one-fourth of the requirement when $\mu = 1$ is used. Moreover, with $\mu = 2$, the computing time is half the time required with $\mu = 1$.

The ranges of a, b and r used by the MHT at the different iterations are shown in Table 2 for an image of size 512×512 containing a circle of radius 210 and having its center at (230,235). $L = 4$, $\sigma = 2$, and $\mu = 2$ have been used. The first iteration uses the full range of the parameters and the range is then logarithmically reduced at the successive iterations. In all the iterations, the algorithm has been successful in selecting the range such that the actual parameter values are almost in the center of the range.

4 CONCLUSIONS

In this paper, we have presented a computationally efficient technique to detect circles in images. Binary images have been used to illustrate the algorithm. It has been shown that the center and radius can be determined to an accuracy of less than 0.2% by using only

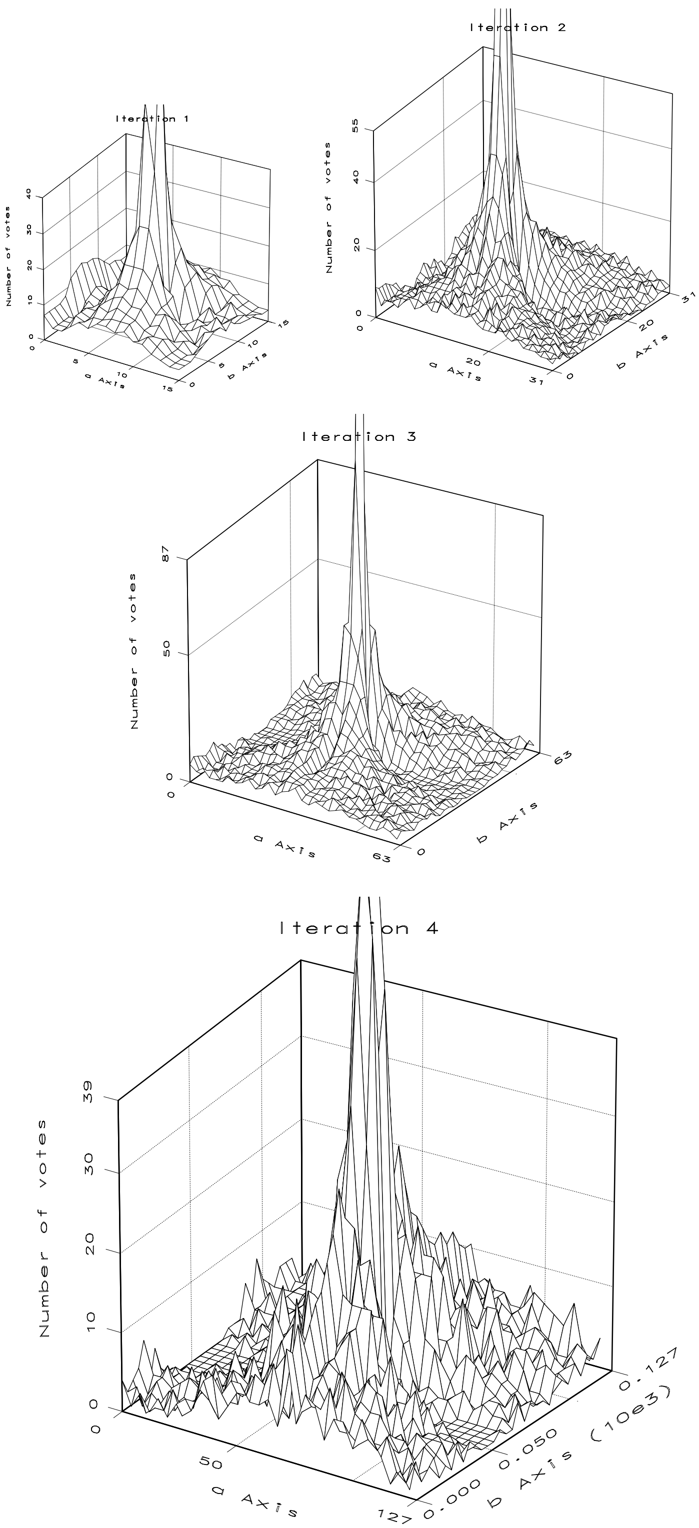

Figure 3: Accumulator array planes during the different iterations.

Table 2: The parameter ranges of investigation at the different iterations for a circle having a radius of 210 and center at (230,235) in a 512×512 image.

Iter. No.	Parameter range of investigation		
	a range	b range	r range
1	0 – 512	0 – 512	0 – 512
2	71.8 – 376.2	103.8 – 408.2	71.8 – 376.2
3	188.2 – 278.8	191.7 – 282.2	169.2 – 259.7
4	222.5 – 236.0	227.4 – 240.9	203.5 – 217.0

four iterations. If prior edge gradient information for the feature points is available, the information can be used to further reduce the amount of computation.

The algorithm illustrated in this paper applies to the detection of circular arcs. With a change in the parametric equation, it can be used to detect ellipses and other parameterizable patterns in images.

References

[1] P.V.C. Hough, "Methods and means for recognizing complex patterns." U.S. Patent 3069654, 1962.

[2] V.F. Leavers, "The dynamic generalized Hough transform: Its relationship to the probabilistic Hough transforms and an application to the concurrent detection of circles and ellipses," *Computer Vision, Graphics and Image Processing: Image Understanding*, vol. 56, no. 3, pp. 381–398, November 1992.

[3] M. Atiquzzaman, "Pipelined implementation of the Multiresolution Hough transform in a pyramid multiprocessor," *Pattern Recognition Letters*, vol. 15, no. 9, pp. 841–851, September 1994.

[4] A. Kavianpour, S. Shoari, and N. Bagherzadeh, "A new approach for circle detection on multiprocessors," *Journal of Parallel and Distributed Computing*, vol. 20, pp. 256–260, 1994.

[5] M. Atiquzzaman, "Multiresolution Hough transform – an efficient method of detecting pattern in images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 11, pp. 1090–1095, November 1992.

[6] J. Illingworth and J. Kittler, "Adaptive Hough transform," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-9, no. 5, pp. 690–698, September 1987.

[7] M.W. Akhtar and M. Atiquzzaman, "Determination of line length using Hough transform," *Electronics Letters*, vol. 28, no. 1, pp. 94–96, January 2, 1992.

[8] M. Atiquzzaman and M.W. Akhtar, "Complete line segment description using the Hough transform," *Image and Vision Computing*, vol. 12, no. 5, pp. 267–273, June 1994.

[9] V.F. Leavers, "Which Hough transform," *Computer Vision, Graphics and Image Processing: Image Understanding*, vol. 58, no. 2, pp. 250–264, September 1993.