Volume 6, Issue 6, June 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

ROI Reduction Approach for Fast Circle Recognition using Classical Hough Transform

Megha Chhabra*
Department of Computer Science and Engineering, Sharda University, Greater Noida, Uttar Pradesh, India

Abstract— Circle Hough transform (CHT) provides a robust technique for iris detection, but the large amount of storage and computational complexity are the major drawbacks of it. The research in this paper proposes a modified CHT which considerably improves the speed of the process without compromising the accuracy of the technique. In the extended work presented in this paper, information of valid region is exploited to develop appropriate method for speedy and correct extraction of circle. To reduce computation time, HT space is divided into equal sized four quadrants which lie in most probable area of the image to find circle center and radius. These quadrants contribute in finding the significant region of the image which needed to be processed. Hence, determining the valid region is the important step towards fast detection. This also permits limiting the memory needs of the algorithm. After that CHT is applied on all four extracted valid regions. The incorrect value detected from the quadrants when compared with allowed deviation threshold from CHT is discarded and correct value is utilized in calculating efficiency of the same over CHT.

Keywords— Iris, Classical Hough transform, accumulator array.

I. INTRODUCTION

As an important part of scientific community, object recognition has become admirably suitable facet of digital image processing. Circular objects are recurrently seen in many images. A large amount of literature exists for circle detection. Iris recognition is one such area which is loaded enough to research into and desires attention and upgrading. It is used in user verification security systems as well as in medical fields.

Iris recognition is based on the fact that the significant information for recognition is found within iris outline which is modeled as circle. Rest of the data from the eye region is redundant. Hough transform (HT) is an effective method for confirming the co-ordinates of the center of the circle and its radius. Circle Hough transform (CHT) provides a robust technique for iris detection, but the large amount of storage and computing complexity are the major drawbacks of it. Many modified versions have been applied by researchers in order to reduce computational complexity and memory requirements. Present work proposes a modified CHT which considerably improves the speed of the process without compromising the accuracy of the technique. On the basis of earlier research and results, time comparison and the efficiency of various methods to detect iris in digital images are compared. An exhaustive analysis is conducted for a large number of iris images.

Edge detection is a potential move in pattern recognition of digital images. The goal of the edge detection process in a digital image is to determine the frontiers of all represented objects. Several algorithms have been proposed to accomplish this task.

Hough transform [1] has long been recognized as a technique of almost unique promise for shape and motion analysis in images containing noisy, missing, and extraneous data but its implementation has been deliberate due to its computational and storage complexity and the lack of a detailed understanding of its properties. The detection of circular and elliptic shapes is a common task of HT. The Hough transform is a well known technique for detecting parametric curves in images. This algorithm has many applications in the real world which makes it a potential area for research. Shearer & kitchen [3] presented an efficient algorithm for filling gaps in the voting procedure for a Hough transform caused by quantization effects. These gaps adversely affect peak detection. This also described improvement of same in any number of dimensions. Walsh & Raftery [4] proposed an algorithm in which Simple clustering methods provide good identification of curve parameters with the removal of noise parameters which is achieved via a simple thresholding of the importance weights. Fernandes & Olivera [8] successfully introduced an improved voting scheme for the HT that allows a software implementation to achieve real-time performance even for relatively large images. The approach clustered approximately collinear edge pixels and, for each cluster, casts votes for a reduced set of lines in the parameter space, based on the quality of the fitting of a straight line to the pixels of the cluster. This improvement not only significantly improved the performance of the voting scheme, but also produced a much cleaner voting map and made the transform more robust to the detection of spurious lines. Cha et al. [5] developed a new extension to the HT by intensifying the Hough space by a third parameter, the horizontal or vertical coordinate of the image space, to provide incremental information as to the length of the linear feature being sought. Therefore, short lines can now be more easily detected. They have also used a Bayesian probabilistic approach that additionally enlarged the precision of extended

Hough transform. Zhao et al. [9] proposed a new algorithm in which the HT is only employed in column-pair spaces of large-scale microarray data, to reduce the computational complexity. Lu & Tan [10] developed an iterative randomized Hough transform (IRHT) for recognition of incomplete ellipses in images with strong noise. The advantage of using IRHT is iterative parameter adjustments and the reciprocal use of the image and parameter spaces.

Shekhar et al. [6] described a novel method to automatically localize the optic disk using HT. A circular region of interest was found by first isolating the brightest area in the image by means of morphological processing, and then the Hough transform was used to detect the circular feature of the optical disk, hence reducing the time of computation. Cheng et al. [18] presented a method in which an eliminating particle swarm optimization (EPSO) algorithm was employed to reduce the time consumption of HT. The parameters of the solution after Hough transformation were considered as the particle positions, and the EPSO algorithm searched the optimum solution by eliminating the weakest particles which speed up the computation. Smereka & Duleba [11] introduced an effective method for circular object recognition like nuclei of cells, which showed robustness for irregularities and for disturbances like noise. In addition to it, Singh et al. [12] conducted an analysis of time complexity on various stages of skew detection process by a preprocessing stage using a simplified form of block adjacency graph (BAG),followed by voting process using the Hough transform and then de-skewing the image using rotation. Gall & Lempitsky [16] introduced the Hough forests approach for object detection by building discriminative class-specific part appearance codebooks based on random forests that cast probabilistic votes within the Hough transform framework. Apart from the accuracy, the use of random forests potentially allows a very time efficient implementation. Du & Yang [13] demonstrated a robust and highly accurate method for detecting the radiation center of a single circular or rectangular field. Method employed the sub pixel accuracy of the HT method which depends on prior knowledge of the field shapes and reduced statistical variance due to averaging information from multiple pixels. Guo et al. [17] presented an idea to suppress the impact of noise edges on accumulation of votes in Hough space produced by complex background or texture regions in images. Surround suppression is employed to allocate different weights to votes of different edges according to the region in which they are located. Peaks formed by noise edges are thus lowered compared with those formed by clear edges, which often give the boundaries between different objects of interest. Maji & Malik [14] introduced the Hough transforming discriminative framework which leads to improved accurateness by allowing the local parts to vote for possible transformations of the object hence allowing use of the peaks of the voting space for importance sampling of windows for further evaluation. Lehmann et al. [27] added a research in literature by producing better detection by using Principled Implicit Shape Model (PRISM) which interprets Hough voting as a dual implementation of linear sliding window detection. It also explained how to avoid soft-matching and spatial pyramid descriptors during detection without losing their positive effect, hence making algorithms simpler and faster. Knopp et al. [22] proposed 3D use of Hough transform. Authors introduced 3D SURF features in combination with the probabilistic Hough voting framework for the purpose of 3D shape class recognition. Bhatia & Chhabra [23] described a method for enhancing the speed of extracting circle i.e. iris from the image by extracting a valid region from the image, using circle Hough transform. This not only reduced the image storage and the quantity of operations, it also improved the speed of the process, without compromising the accuracy of the technique. A modified version of Hierarchical Hough transform was also proposed which further reduced the time taken to detect iris. Borrmann et al. [20] proposed a novel accumulator design, whose cells were of equal size. This property leads to an easier detection of planes when using the Hough Transform. The proposed accumulator is compared to previously known designs as well. Tu et al. [25] discovered the self-similarity in HT butteries. Based on this property a simple method was proposed to obtain a very high resolution HT without the limitations associated with peak splitting and vote spreading. Hough transform has been accompanied with many techniques in order to improve the results to be followed from the intended application.

Reina et al. [7] proposed an algorithm that used a robust Hough transform enhanced by fuzzy reasoning to estimate the angle of inclination of the wheel trace with respect to the vehicle reference frame. In addition to it, Izadinia et al. [19] proposed a new method, namely Fuzzy Generalized Hough Transform (FGHT), in which a set of fuzzy rules were referred by the gradient direction of edge pixels and vote for the probable position of the center. Additionally, the proposed method could identify the boundary of the rotated and scaled object via a new voting strategy with least error under various conditions. Mokhayeri & Akrbarzadeh [26] proposed a system in which the eye region is detected using the genetic algorithm (GA), and a fuzzy filter is designed for noise reduction. Edge detection was performed based on fuzzy reasoning and linking was done using Hough transform. Then relevant features extracted of an eye using signal processing technique, were imported into the learning system to classify the active states between stress and relaxed condition of eye. Fuzzy systems are not preferred with HT when the application province is time consuming.

The paper is organized as follows. In section 2, we describe existing method to detect circles. In section 3 , we define the proposed CHT to extract it in efficient way and the mathematical model of the proposed CHT. Section 4 consists of results and comparative analysis. Section 5 contains the concluding remarks about the proposed experiment.

II. CLASSICAL HOUGH TRANSFORM

Detecting iris (circle) using CHT includes simple Hough Transform method. CHT is easy to understand but takes large computational time.

Following are the steps to detect circle using Classical Hough Transform:

Step1: In order to detect circle, equation of a circle is given by:

$$(x-h)^2+(y-k)^2=r^2 \qquad (1)$$

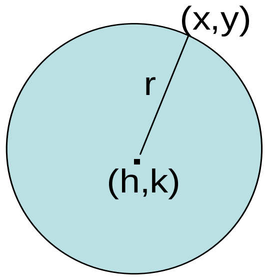


Fig1: Circle with center (h, k) and radius r.

Where (x, y) is set of all points on circle, r is radius and (h, k) is center of circle.

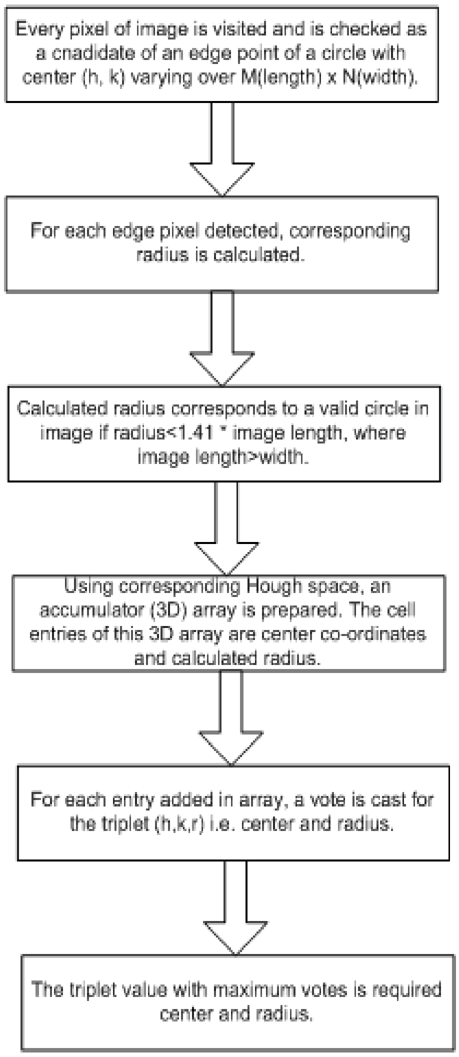


Fig2: Graphical representation of Classical Hough Transform

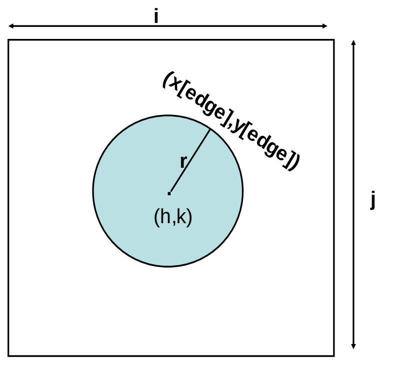


Fig3: An image representing relation between edge points and image length and width.

With a given gray scaled image $f(x,y)$ of size $M \times N$, every pixel of it is visited and is checked as a candidate of an edge point of a circle with center (h,k) varying over MN pixels using sobel operator.

Step 2: For each edge point in image, corresponding radius is calculated.

$$r = sqrt((i - xedge[t])^2 + (j - yedge[t])^2) \quad (2)$$

Where
$0 < t <$ number of edge points,
$0 < i <$ image width,
$0 < j <$ image length

This radius must lie inside edges of image. Using the values of corresponding center co-ordinates and radius, an accumulator array [3D] is prepared as fig 3.

Step 3: In this 3D array, cell values contain accumulated votes for each center coordinate and corresponding radius generated. This calls for a circle in the image as required circle (or iris) which contains maximum value in array. Using every value of pixel of image and edge points, the radii calculated gives a peak. Apart from local maxima, the required radius with maximum votes lies at global maxima and that value is used as center co-ordinates and radius as (h_0, k_0, r_0) of required circle (iris).

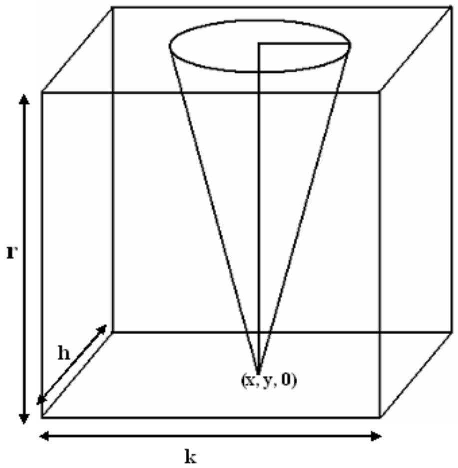
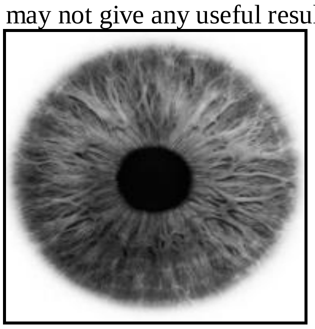


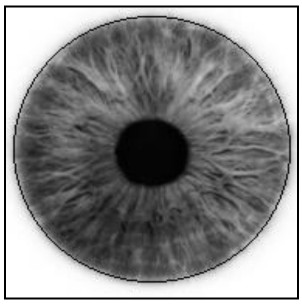
Fig4: Hough space for CHT

As presented in fig 4, (a) is input image to CHT and (b) is the output of CHT as the detected iris. CHT uses every pixel of image and that makes it store almost every pixel and process it. This results in easy processing of image. Also it results in large computational time for detection and large memory storage area for data to be processed for same.

So, CHT is quite easy in implementation whereas takes large computational time and storage area. In other words, CHT visits pixels which may not give any useful results.



(a)



(b)

Fig5: (a) Captured image of iris, (b) Detected iris from image

III. MODIFIED CIRCLE HOUGH TRANSFORM

In case of HT based detection methods execution speed, hence time, is directly proportional to the number of edge pixels in the input image. The proposed method reduces the number of image edge pixels to be processed. Reduction in time of execution of programs to extract circle from image is the basic and probably most important feature of this work. The motive is achieved in two steps:

- Limiting the image space to the valid region.
- Apply transforms on the valid region extracted.

Further, limiting the image space is achieved in following steps:

- Transform the image into HT space.

Divide the HT space into four quadrants. These quadrants are of maximum probability to find suitable and accurate results.

Determining valid region to process out of whole image space $f(x,y)$ provided over $\text{length}(M)\times\text{width}(N)$ is first and an important step. Presently, four equal sized quadrants are constructed in HT space as shown in Fig 5. The purposes of these quadrants are:

- To use most probable areas of image from where center and radius of iris angled or inclined anywhere in the image can be found out in faster way.
- To reduce number of pixels to process.

This work comprises of analysis of time of execution of this improved transform and comparison of same with existing CHT. The basic idea behind the use of relatively smaller area in HT space of CHT transform is to analyze the fact that without compromising the accuracy of detection of iris and reducing the number of pixels to visit, whether it still performs efficiently or not. Fig 6 shows the algorithm of modified CHT.

With the aim of improving time of existing CHT, the first step is devoted to obtain the valid region to process.

Following are the steps performed as preprocessing of CHT:

Step 1: As shown in Fig 5, the number of pixels to be processed is reduced using only four equal sized quadrants for finding center and radius of circle (iris).

Step 2: Let $f(x,y)$ be the image of size $M\times N$. Let A is the area of $f(x,y)$ then it is given by A = $M\times N$. Divide f(x, y) in four equal quadrants, such that, 1st quadrant's rectangular boundary is $((M/4,N/4),(M/2,N/2))$, 2nd quadrant's rectangular boundary is $((M/2,N/2),(3M/4,N/2))$, 3rd quadrant's rectangular boundary is $((M/4,N/2),(M/2,3N/4))$, 4th quadrant's rectangular boundary is $((M/2,N/2),(3M/4,3N/4))$.

Step 3: Now transformation is applied on each quadrant. Since edge points are now distributed into four quadrants, hence all the quadrants are evaluated for triplet value and which contains lesser number of edge points will result in faster determination of triplet (h0, k0, r0).

There comes a set of 4 triplets from 4 quadrants. Each value is compared on the basis of radius and value of triplet starting with smallest radius is used to draw detected circle.

As clear from Fig 7, the valid regions are these quadrant with each having area 16 times lesser than original image. If A' is area of each quadrant then it is given by

$$A' = \frac{M}{4}\times\frac{N}{4} = \frac{A}{16}$$

Step 4: Now CHT can be performed on each such quadrant. The incorrect value detected from the quadrants when compared with allowed deviation threshold from CHT is discarded and correct value is utilized in calculating efficiency of the same over CHT.

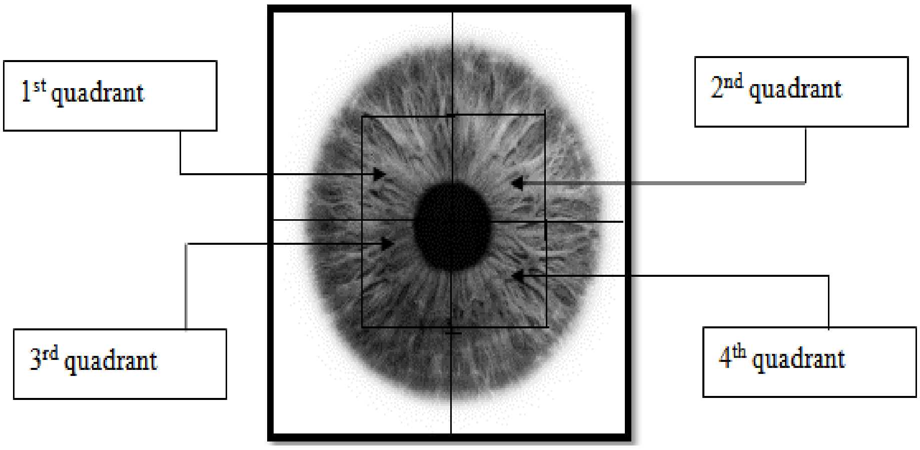


Fig 6: Equal sized quadrants.

Global maximum is divided among 4 global maxima from 4 quadrants. Maxima from any quadrant may shift to other quadrant due to various reasons like glare in any particular quadrant(s), inclination of iris. This may result in false local maxima which may give output as radius with false maximum votes in that quadrant. In order to reduce this risk, all 4 values are compared and value repeating with maximum votes out of all quadrants votes and with minimum deviation from CHT triplet is considered as detected optimal value. Its correctness can be seen from drawn circle on eye using any circle drawing algorithm.

Divide f(x, y) in four equal quadrants, such that

- 1st quadrant's rectangular boundary is ((M/4, N/4), (M/2, N/2)).
- 2nd quadrant's rectangular boundary is ((M/2, N/4), (3M/4, N/2)).
- 3rd quadrant's rectangular boundary is ((M/4, N/2), (M/2, 3N/4)).
- 4th quadrant's rectangular boundary is ((M/2, N/2), (3M/4, 3N/4)).

↓

Now, Classical Hough Transform is applied on each quadrant.

↓

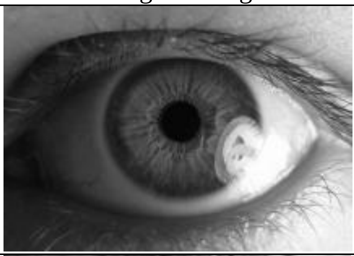
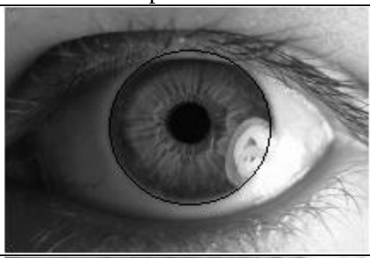
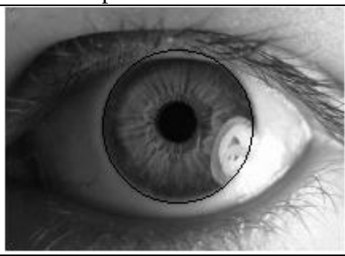
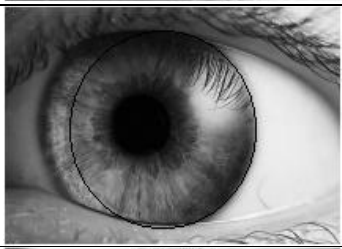
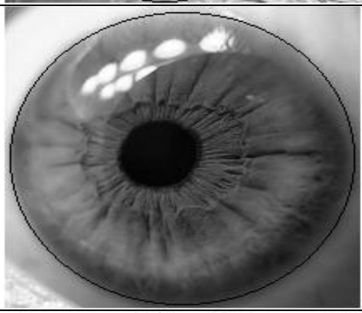
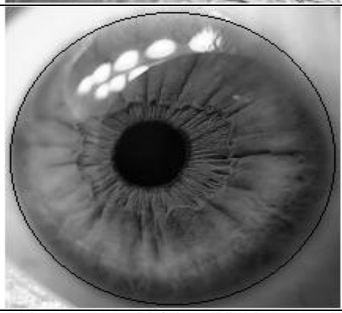
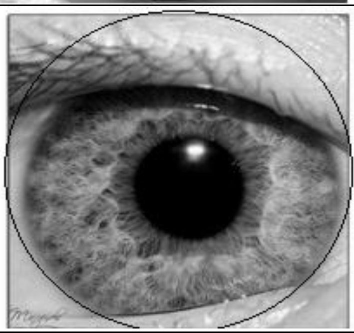
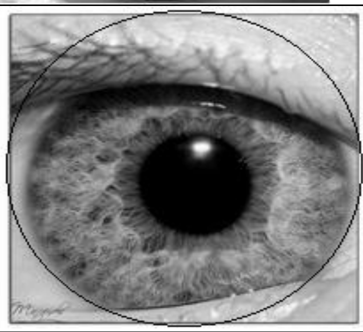
Valid regions are the quadrant with each having area 1/6 times the original image. Hence processing time for original image reduces by 4 times.

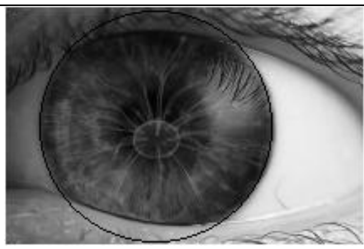
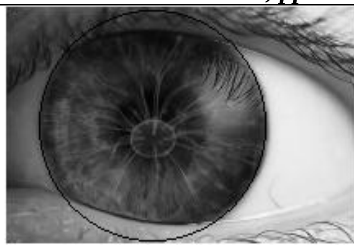
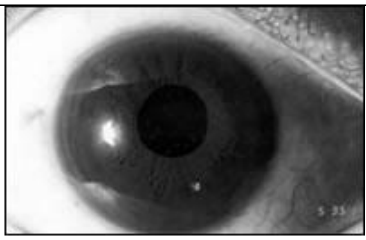
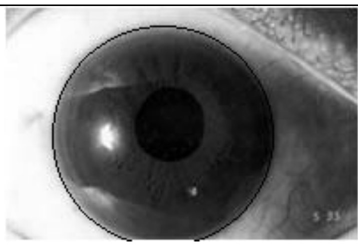
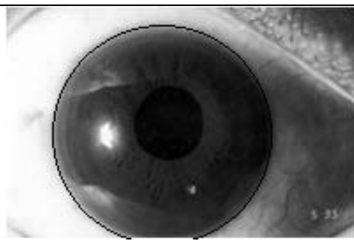
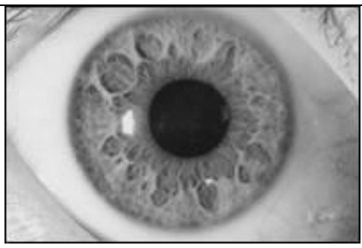
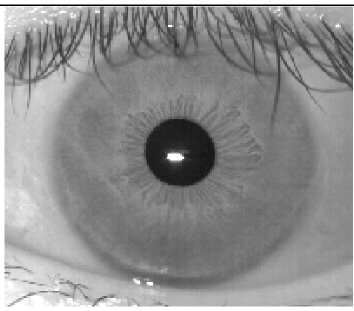
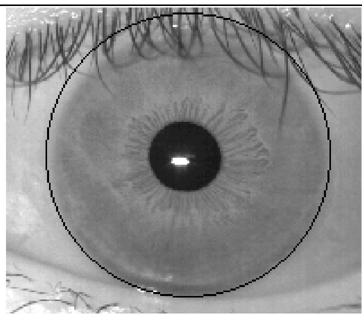
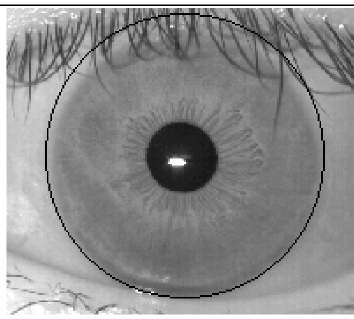
↓

The quadrant with lesser number of edge pixels will result in faster determination of triplet (h_0, k_0, r_0).

Fig7: Graphical representation of Modified Classical Hough Transform

Table I Analysis of detected iris using CHT and MCHT

Sno.	Original image	Output of CHT	Output of modified CHT
1			
2			
3			
4			

5			
6			
7			
8			
9			

Assume an image containing iris, divided into four equal quadrants (Fig 7). This apparently reduces computational time and memory requirements. From that change of area, number of indices of accumulator array decreases and hence lesser no. of peaks or no. of votes to choose from also decreases.

From above, it's concluded that:

- Computationally fast.
- Less memory requirements.
- Easy to understand and implement.
- Recognition of valid region.

Area to visit decreases by almost a factor of 16 for each time CHT runs.

IV. RESULTS AND COMPARISON

This work comprises of the comparative study of the proposed algorithm and existing CHT. This will provide an in-depth analysis of the fact that the proposed method is a better recognition algorithm than already existing CHT. To validate the script, experiment has been conducted on 100 images of human eye. The results of detected iris from the collected are shown in Table 2. The results so obtained are not all up to 100% accuracy. There are outputs which detect wrong center and radius. These inaccurate evaluations are caused by presence of eyelids in the image, angular displacement of iris, appearance of glare in the image captured etc.

A. Analysis of detected iris by CHT and MCHT

Table1 shows the pictographic analysis of detected iris by CHT and MCHT with original image of the iris.

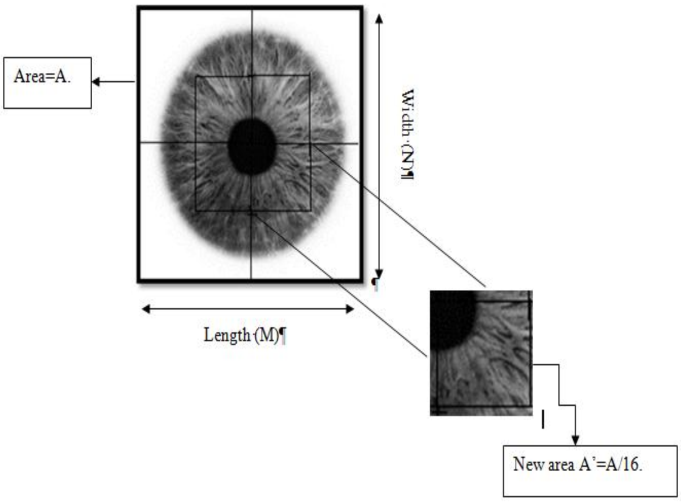


Fig8: Reduction in area to be processed in Modified Classical Hough Transform.

B. Analysis of time of execution of CHT and MCHT

table 2 shows the efficiency of proposed method over existing CHT by comparing time of execution of both CHT and MCHT. Maximum efficiency show over an image is 96.64174252%. Table 3 shows comparison of the radius and center calculated by two methods. Numbers bold are the images which differ in result.

C. Analysis of graphical comparison between CHT and MCHT

Fig 8 shows the graphical representation of efficiency of CHT and modified CHT. It clearly shows the difference in peaks of time of execution of CHT and modified CHT.

D. Comparison and discussion

If the input image Fig 9 is say img_7 and the output image Fig 10 say img_7_cht is the image showing result of circle detection by CHT then Fig 12shows that the time taken by CHT to detect circle center and radius is 33.983 seconds. It's clear from the Fig 13 that time taken by modified CHT is 2.45 seconds. This shows an efficiency of 93.95413816%. Also, the results show that both detected the same center and radius co-ordinates. If img_7.tif is input image, then output of modified CHT is:

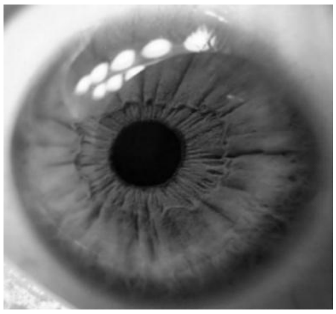


Fig 8: Sample input image img_7.tif.

On comparing CHT with modified CHT, it clearly shows a great amount of reduction in execution time. Hence, it shows the maximum improvement of about 97.64174252%. On an average, its performance increases by 94.45732238%.

Table 5 shows various differences between existing and proposed CHT on the basis of time to execute, space required to store data, and implementation complexity.

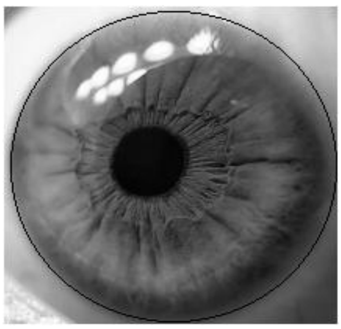


Fig9: Sample output image img_7_cht.tif.

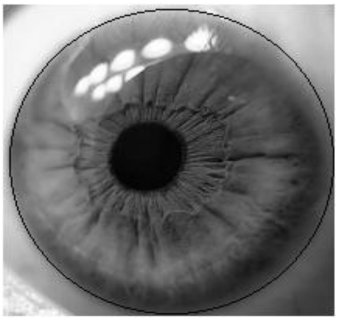


Fig10: Sample output image img_7_mcht.tif.

V. CONCLUSIONS

Modified CHT for detection of circle from digital images suitable for iris images are proposed. A complete analysis of classical method and proposed method is obtainable. HT based algorithms are time consuming and use quite large memory for data storage. In addition to it, execution speed of HT based detection methods is directly proportional to the number of edge pixels in the input image. The proposed method reduces the number of image edge pixels to be processed at a time using the smaller quadrants hence increasing the efficiency of detection as well. The gain in efficiency does not affect the robustness of the techniques. These advantages make the algorithms most suitable for applications where iris location detection is a critical aspect. To validate the script, a data set of 100 images of human eye is evaluated. The results of detected iris from the collected data set are shown in Table 2. The results so obtained are not all up to 100% accuracy. There are outputs which detect wrong center and radius. These inaccurate evaluations are caused by presence of eyelids in the image, angular displacement of iris, appearance of glare in the image captured etc.

Table 2: Comparison of CHT and MCHT.

NAME OF METHOD	COMPUTATIONAL TIME	SPACE REQUIREMENTS	IMPLEMENTATION COMPLEXITY
CHT	**HIGH**	**HIGH**	**MODERATE**
MCHT	**LOW**	**LOW**	**LOWER**

REFERENCES

[1] R. Duda and P. Hart, Use of Hough transform is to detect lines and curves in pictures, Communications of the ACM 15, (1975), 11-15.
[2] S. Tsuji, F. Matsumoto, Detection of ellipses by a modified Hough Transformation, IEEE Transactions on Computers, (1978) 777-781S.
[3] K. Shearer, L. Kitchen, Filling gaps in the Hough-transform voting locus for N-dimensional parameter spaces, Pattern Recognition Letters 16 (1995) 1105-1109.
[4] D. Walsh, A. Raftery, Accurate and efficient curve detection in images: the importance sampling Hough transforms, Pattern Recognition, (2002), 1421-1431.
[5] J. Cha, R.H. Cofer, S.P. Kozaitis, Extended Hough transform for linear feature detection, Pattern Recognition, (2006), 1034-1043.
[6] S. Sekhar, W. Al-Nuaimy A.K. Nandi, Automated localisation of retinal optic disk using Hough transform, 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 1577-1580.
[7] G. Reina, G. Ishigami, K. Nagatani, K Yoshida, Vision-based estimation of slip angle for mobile robots and planetary rovers IEEE International Conference on Robotics and Automation (2008). 486-491.
[8] L. Fernandes, M. Oliveira, Real-time line detection through an improved Hough transform voting scheme, Pattern Recognition, (2008), 299-314.
[9] H. Zhao, A. W. Liew, X. Xie, H. Yan, A new geometric biclustering algorithm based on the Hough transform for analysis of large-scale microarray data, Journal of Theoretical Biology, (2008), 264-274.
[10] W. Lu, J. Tan, Detection of incomplete ellipse in images with strong noise by iterative randomized Hough transform (IRHT), Pattern Recognition, (2008), Pages 1268-1279.
[11] M. Smereka and I. Duleba, Circular object detection using a modified hough transform, Int. J. Appl. Math. Comput. Sci., (2008), 85-91.
[12] C. Singh, N. Bhatia, A. Kaur, Hough transform based fast skew detection and accurate skew correction methods, Pattern Recognition(2008), 3528-3546.
[13] W. Du, J. Yang, A robust Hough transform algorithm for determining the radiation centers of circular and rectangular fields with subpixel accuracy, Phys. Med. Biol. (2009) 555-567.
[14] S. Maji, J. Malik, Object Detection using a Max-Margin Hough Transform, Computer Vision and Pattern Recognition, (2009) 1038-1045.

[15] R.Bansal, P. Arora, M. Gaur, P. Sehgal, P Bedi, Fingerprint Image Enhancement Using Type-2 Fuzzy Sets, Sixth International Conference on Fuzzy Systems and Knowledge Discovery, (2009)., 412-417.
[16] J. Gall, V. Lempitsky, Class-specific Hough forests for object detection, IEEE Conference on Computer Vision and Pattern Recognition, (2009), 1022-1029.
[17] S. Guo, T. Pridmore, Y. Kong, X. Zhang, An improved Hough transform voting scheme utilizing surround suppression, Pattern Recognition Letters, (2009), 1241- 125.
[18] H.D. Cheng, Y. Guo, Y. Zhang, A novel Hough transform based on eliminating particle swarm optimization and its applications, Pattern Recognition, (2009), 1959-1969.
[19] H. Izadinia, F. Sadeghi, M. Ebadzadeh, Fuzzy generalized hough transform invariant to rotation and scale in noisy environment, IEEE International Conference on Fuzzy Systems FUZZ-IEEE (2009), 153-158.
[20] D. Borrmann, J. Elseberg, K. Lingemann, A. Niichter, A Data Structure for the 3D Hough Transfrom for Plane Detection, 7th IFAC Symp. Intell. Aut. Vehicles (2010).
[21] H. Ruppertshofen, C. Lorenz, P. Beyerlein, Z. Salah , G. Rose, H. Schramm Fully automatic model creation for object localization utilizing the Generalized Hough transform, Bildverarbeitung fr die Medizin. Springer, (2010) 281-285.
[22] J. Knopp, M. Prasad, G. Willems, R. Timofte, L. Van Gool, Hough transform and 3d surf for robust three dimensional classification, ECCV, (2010).
[23] N.Bhatia, M. Chhabra, Improved Hough transform for fast iris detection, Proceedings on International Conference on Signal Processing Systems, Dalian (2010), 172-176.
[24] A. Maki, F. Perbet, B. Stenger, Demisting the Hough transform for 3D shape recognition and registration, BMVC, 2011.
[25] C.Tu, S. Du, B.J. Van Wyk, K. Djouani, Y. Hamam, , High resolution Hough transform based on buttery self-similarity, Electronics Letters , (2011),1360-1361.
[26] F. Mokhayeri, M.R Akbarzadeh, Mental Stress Detection Based on Soft Computing Techniques, IEEE International Conference on Bioinformatics and Biomedicine (BIBM), (2011), 430-433.
[27] A. Lehmann, B. Leibe, LV. Gool , Fast prism: Branch and bound Hough transform for object class detection, International Journal of Computer Vision, (2011), 175-197.
[28] T.Y. Lee, T.S. Chang, S.H. Lai, K.C. Liu, H.S. Wu, Wide-angle distortion correction by Hough transform and gradient estimation, IEEE Visual Communications and Image Processing (VCIP), 2011, 1-4.

Table 2 Comparison of time of execution of CHT.

Name	Size	CHT [1] time(in sec)	MCHT time(in sec)	Efficiency of MCHT over CHT (%)
img_1	215x153	10.392	0.803	92.27290223
img_2	215x154	10.466	0.803	92.32753679
img_3	215x143	16.222	0.981	93.95265689
img_4	215x156	12.909	0.961	92.55558138
img_5	215x158	41.516	2.51	93.95413816
img_6	215x215	18.686	1.285	93.12319383
img_7	215x225	9.796	0.745	92.39485504
img_8	215x137	8.92	0.682	92.35426009
img_9	215x144	8.99	0.696	92.25806452
img_10	215x146	27.392	1.814	93.3776285
img_11	215x143	23.165	1.55	93.30887114
img_12	215x214	5.653	0.465	91.77427914
img_13	215x205	26.098	1.696	93.50141773
img_14	215x162	3.934	0.301	92.34875445
img_15	215x214	3.927	0.358	90.88362618
img_16	215x143	12.83	0.898	93.00077942
img_17	256x171	27.76	1.811	93.47622478
img_18	256x171	15.353	0.725	95.27779587
img_19	215x215	44.814	2.702	93.97063418

img_20	215x144	50.301	1.587	96.84499314
img_21	256x224	82.179	2.422	97.05277504
img_22	215x215	79.485	3.238	95.9262754
img_23	256x178	59.733	1.816	96.95980446
img_24	256x200	20.113	0.695	96.54452344
img_25	256x208	17.902	0.633	96.46408223
img_26	256x208	15.504	0.575	96.29127967
img_27	256x255	23.973	0.806	96.63788429
img_28	215x214	31.652	1.027	96.75533932
img_29	256x192	31.41	1.028	96.72715696
img_30	194x200	19.994	0.688	96.55896769
img_31	203x175	32.415	1.047	96.77001388
img_32	200x173	25.8	0.857	96.67829457
img_33	247x164	23.728	0.793	96.65795684
img_34	247x168	18.108	0.65	96.41042633
img_35	200x150	40.987	1.279	96.87949838
img_36	247x185	23.166	0.784	96.61572995
img_37	200x133	16.992	0.614	96.38653484
img_38	200x134	14.25	0.534	96.25263158
img_39	247x204	22.762	0.781	96.56884281
img_40	247x186	23.87	0.796	96.66527021
img_41	250x170	40.761	1.28	96.85974338
img_42	247x165	17.245	0.634	96.32357205
img_43	215x161	14.321	0.536	96.25724461
img_44	200x150	51.159	1.666	96.74348599
img_45	200x150	79.914	2.541	96.82033186
img_46	215x143	86.35	3.249	96.23740591
img_47	247x165	57.918	1.904	96.71259367
img_48	215x177	17.307	0.698	95.96694979
img_49	215x162	21.639	0.576	97.33813947
img_50	215x146	30.76	1.033	96.64174252

Table 3 Comparison of the radius and center of iris calculated by CHT and MCHT.

Name	size	CHT [1]		MCHT	
		Center	radius	Center	radius
img_1	215x153	95,78	64	**98,74**	**60**
img_2	215x154	95,74	64	**94,70**	**65**
img_3	215x143	95,78	64	**95,78**	**65**
img_4	215x156	107,108	104	**107,108**	**105**
img_5	215x158	107,108	99	107,108	99
img_6	215x215	99,61	55	99,61	55
img_7	215x225	90,68	70	90,68	70
img_8	215x137	111,78	65	**111,72**	**68**

img_9	215x144	111,75	62	111,75	62
img_10	215x146	110,106	89	110,106	89
img_11	215x143	110,102	85	110,102	85
img_12	215x214	134,97	43	134,97	43
img_13	215x205	110,106	89	110,106	89
img_14	215x162	126,96	70	135,83	85
img_15	215x214	126,96	70	126,96	70
img_16	215x143	143,87	46	169,83	60
img_17	256x171	129,108	97	129,108	97
img_18	256x171	108,69	65	108,69	65
img_19	215x215	128,92	70	128,92	70
img_20	215x144	109,112	97	109,112	97
img_21	256x224	124,84	70	124,84	70
img_22	215x215	131,128	106	131,128	106
img_23	256x178	110,106	89	110,106	89
img_24	256x200	113,91	87	113,91	87
img_25	256x208	95,103	89	95,103	89
img_26	256x208	106,84	80	106,84	80
img_27	256x255	94,91	89	94,91	89
img_28	215x214	126,85	72	126,85	72
img_29	256x192	128,90	76	128,90	76
img_30	194x200	112,77	70	112,77	70
img_31	203x175	114,86	82	114,86	82
img_32	200x173	135,60	75	135,60	75
img_33	247x164	137,84	82	137,84	82
img_34	247x168	96,78	65	96,78	65
img_35	200x150	123,79	78	123,79	78
img_36	247x185	116,95	101	116,95	101
img_37	200x133	124,51	72	124,51	72
img_38	200x134	119,70	72	119,70	72
img_39	247x204	90,72	68	90,72	68
img_40	247x186	137,84	82	137,84	82
img_41	250x170	123,79	78	123,79	78
img_42	247x165	124,51	72	124,51	72
img_43	215x161	119,70	72	119,70	72
img_44	200x150	109,112	97	109,112	97
img_45	200x150	124,84	70	124,84	70
img_46	215x143	131,128	106	131,128	106
img_47	247x165	110,106	89	110,106	89
img_48	215x177	95,103	89	95,103	89
img_49	215x162	141,141	140	141,141	140
img_50	215x146	126,85	72	126,85	72

