A new Approach for Detection and ExtractionTables in Scanned Document Image using Improved Hough Transform

Dr. Hasanen S. Abdullah
Computer Sciences Department, University of Technology/Baghdad.
Email: qhasanen@yahoo.com
Ammar H. Jasim
College of Science for women, University of Baghdad/ Baghdad.
Email: ammar_hussein_2004@yahoo.com

Received on: 20/1/2016 & Accepted on: 19/5/2016

ABSTRACT

In this paper, an improvement approach of Hough transform for tables detection and extraction from scanned document images is achieved as one of the main stages in document recognition to recognize between original and faked documents. The improvements fundamentally originate from the modulation of the standard Hough transform (SHT)parameter selection, peak threshold and voting scheme. Peakvalues formed by noise edges are thus lowered compared with those formed by clear edges. Experimental results show the proposed method leads to significant suppression of false peaks in Hough space and is thus effective. The parameters can be determined empirically in advance, which their advantage to use the proposed method in fully automated lines detection applications.

Keywords: Hough transforms**,** Document Image Analysis (DIA), Table detection, Table extraction.

INTRODUCTION

Documents can contain tables, which can be placed everywhere within a document page, and have a regular structure [1]. The detection and analysis of tables are a part of current Document Image Analysis (DIA) systems. The table in document image considers structured objects that show relational and statistical information. Detection and extraction of tables from scanned document images become one of the most research and development topics in scanned image processing, Optical Character Recognition (OCR) system and digital library system.

The detection of tables is of interest, since tables normally contain specific data. The occurrence of fast computers, large computer memory, and low-cost scanners encourage rising interest in DIA [2]. DIAhas become gradually more important tools in the automation of office documentation tasks; figure (1) illustrates the DIA categories.Document scanners such text readers and OCR systems are the key component of systems capable of to do these tasks. Nowadays, documents more and more create on the computer. Therefore, the research work results in document analysis and OCR can be seen every day.

https://doi.org/10.30684/etj.34.5B.14
2412-0758/University of Technology-Iraq, Baghdad, Iraq

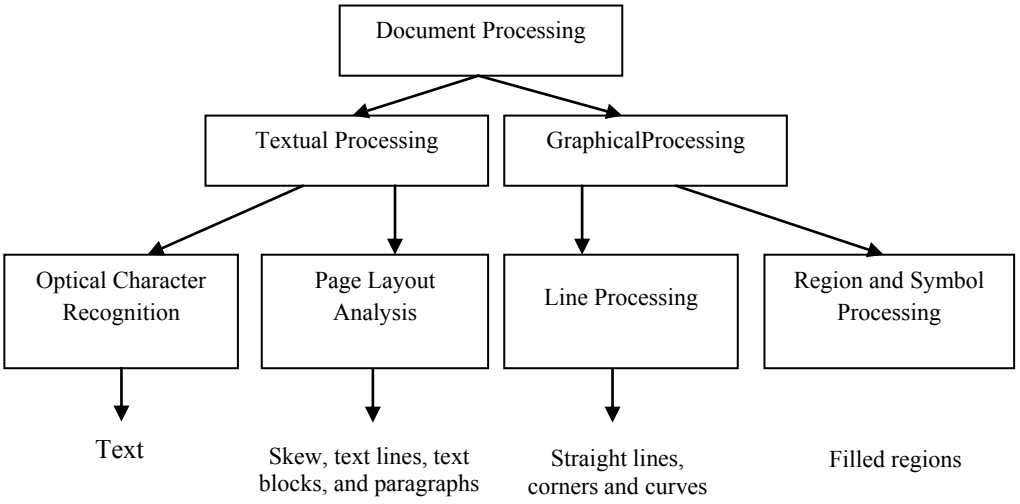


Figure (1), Categories of DIA

Nowadays tables are common and exist in almost type of documents like in books, newspapers... etc. Tables make information easier to understand and realize than regular text block and their ability to describe statistical and relational information, which plays a significant role in DIA [3].

When we need to represent information, Tables are used to represent it in an efficient way, according to the type of information that needed to be represented, the table layout will differ and altar. A variety of formats of the table makes it hard for OCR engine to detect and extract as an Image block.

In this paper, we deal with printed table not handwritten tables and many methods proposed for this purpose to detect and extract tables from document image. Many of this proposed method are base on the Hough transform, Fourier transform, Nearest Neighbor, mathematical morphology and Projection profile.

In this work, the proposed approach is simple but powerful approach for detection and extraction tables from scanned documents. Depending on the notation, that has been observed in the table, the tables have an intersection line, the distances among distinct columns are considerably larger than the distances among the words and the lines of the table be prominent and visible. This deceptive observation allows us to design a simple and powerful table detection and extraction system with low overhead computation and memory cost. A small subset of prominent edge points in an N x N digital image is examine using threshold technique to isolate more notable edge feature. Each pixel of the document image is scanned, and edge information is used to compute the Hough Transform of the image and select a single best maximum highest peak from several transform peaks rather than finding many peaks in transform (Hough) image. Only one peak has to be select from many peaks belong to lines in transforming space. The algorithm starts with very high peak, specify it and select the pixels that participate in it. The proposed approach allows new transform scanned documents to be found without to recalculate the full Hough transform space. This phase is performed in sequence, omit small peaks when large peaks are detected. Therefore, the search space of the HT will be reduced hence the computation time will decrease, while silent maintaining detection performance at a level similar to that of the full Hough transform.

Related Works

There are many papers concentrate on table detection and the following a description of the important one of these literatures. [4]Proposed an approach that can detect all types of table's format from the single column image document based on projection profile and Hough method for line detection. Cesarini et al. [5] illustrate a method for detecting the location of table in scanned document images based on searching for parallel lines in X-Y tree to detect table of the document. Shafait and Smith [**6**] proposed an algorithm to locate tables in documents having a large variety of layouts. They first identify text column partitions that could belong to a table region, referred to as table partitions, group table partitions into table columns and use the horizontal ruling to locate the table. In their paper, they are trying to bridge the column gap. Their goal is to detect table regions in heterogeneous documents. [7] proposed an algorithm that is analyzing the structure of individual pages of a document by detecting chunks of text, and determine the areas in which figures or tables presented by reasoning about the empty regions within that text. In [8] they have presented the table recognition literature from the viewpoint that table recognizers may be understood as sequences of decisions (inferences) supported by observations and transformations of available data. In [9] this paper, they propose a different technique for automatic table detection in document images that neither requires any training phase nor uses domain-specific heuristics. A novel technique for vertical and horizontal line detection in document images is proposed. The technique is mainly base on vertical and horizontal black runs processing as well as on image/text areas estimation in order to exclude line segments that belong to these areas.

Documents Image Analysis

The DIA consists of several stages, which are used as they are needed in any approach in image processing model. Sections bellow describe the main image analysis stages that are used and needed in the proposed system.

Pre-processing

Preprocessing involves image binarization and enhancement it is an important step to feature extraction. Therefore, it controls the convenience of the results for the consecutive steps before going on the line and table detection steps. Thus, the use of preprocessing techniques may enhance table detection that preparing it for the next stage in extraction stage [10].
This preprocessing step is used to reduce the effects of blurred edge transitions and other sources of noise, which are common in scanned images.

Edge Detection

Edge Features consider the most popular features and a very important component in different areas of image analysis; it has reduced the amount of information in the image while kept the important, characteristic and structural information for applications that need to speed up its implementation regardless of accuracy [11, 12]. Therefore, proposed system relies on completing basic edge information using canny edge detection and work on a more accurate edge feature (Prominent edges) while keeping a relatively low computation cost. Nevertheless, the output from an edge filter is still an image described by its pixels.
The goal of using the canny edge detection is to make best possible thin/narrow edges by using non-maximal suppression.

The proposed System Structure

The proposed system is build using several sequential stages that can be described in the section below. Figure (2) will show the main stages of the proposed system.Where L represents an assumed line length in pixels.

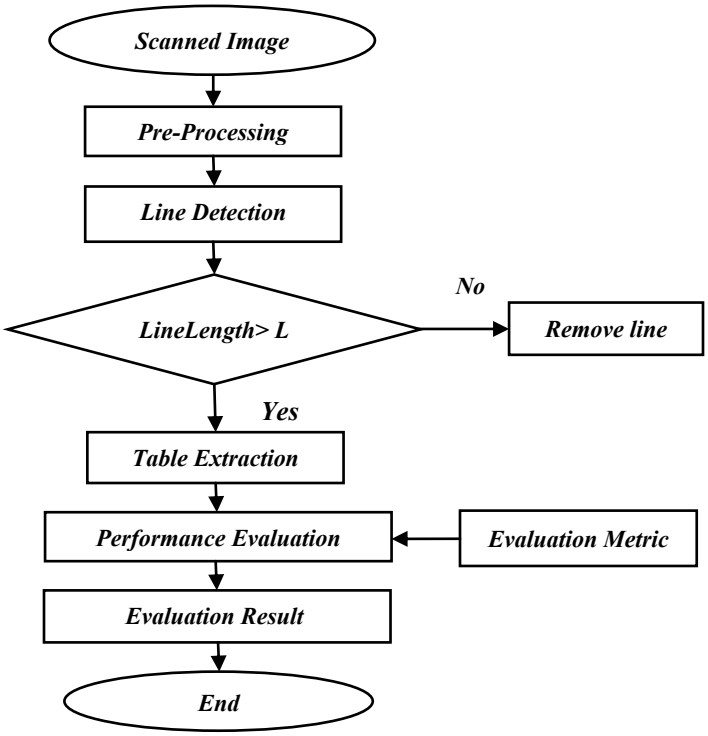


Figure (2), Table Recognition of proposed System Model

In the proposed system, the detect intersection lines that form table for both vertical and horizontal straight lines using improved Hough transform (IHT)(as it is described in section 5); which will assist to detect tables and as a result these tablescan be extracted out of entire document, hence, leading to tables analysis process.

In the next sections, we show the proposed table detection and extraction method for document images using IHT; in addition, the experimental results show the efficiency of the proposed method.

Proposed ImproveHough Transform (IHT) Algorithm

It is necessary to pre-process input images to avoid influence of troublesome effects and carefully chosen parameter space. After that, edges and lines detection can be implemented successfully.

The assumption design to execute the proposed IHT algorithm is to quantize the Hough space by using an array of two dimension for counters votes, where the coordination of this array is represented by the parameters ρ , θ of the line, where ρ is the distance from the (0,0) as a line along a vector vertical to the detected line and θ is the angle of the vertical projection from the (0,0) to the detected line. This array is known as an accumulator array.

Hough transforms working only in parameter space, therefore, the Hough transform limits from high computation complexity. If the size of the image that is processed is large, the number of edge points mapped to parameter space lead to increase this space noticeably as well as increase memory cost and longer processing time for detecting lines.

To overcome the shortcoming of the Hough approach of intensive computations (1-to-N mapping) and the storage requirements (voting space and entry number), the feature points

of the table that are prominent and clear it take into consideration, therefore only deal with a small portion pixels Compared with the overall pixels in an image, hence, it is possible to detect the lines that construct the tables. Moreover, using the thresholding process on parameter space to specify the large points set such as edge pixels and utilize the image space to overcome limitations such as memory size and speed of implementation.

The mathematical basis of the proposed algorithm is also determined to detect lines using Hough Transform with some adjustments in dealing with values of Hough peak and edges of the image. The tables are extracted directly from the document by select peaks from accumulator array for Hough image according to intensity region and then calculate the line's equation applying Hough Transform. This approach allows the algorithm to use the prominent edge information in the image to correlate the edge pixels for constructing the lines of the connected edge pixels and the edge informant that are not intelligible will not take in consideration for fast detection of lines. Start-point, end-point and direction of each detected line are used to detect table construction, and pairs of intersected lines are used to form table structures. A table is detected when the extracted peaks value achieves specific geometric conditions.

The experiments show that the proposed IHT-based table recognition method is well organized and accurate, particularly for large images.Each pixel of the input document image is scanned and computes the Hough Transform of a small set of edge pixel in the image. The searching priority of peak detection is set according to the peak value.The peaks of the Hough space (which belong to lines) are extracted, and table rows and columns are detected when four extracted peaks satisfy certain geometric conditions according to orientation observe using the information of gradient direction to direct the voting process.

The Standard Hough Transform uses the polar representation of a line:

$$\rho = x*\cos(\theta) + y*\sin(\theta) \quad \text{... (1) [13, 14]}$$

Where ρ is normal distance between the origin and the line and θ is normal angle of line from origin[13]. The adapted range of θ is −90°,+90° for vertical line and θ=0, 180° for a horizontal line. The angle of the line itself is θ+90°, also measured clockwise with respect to the positive *x*-axis. Figure (3) shows the vertical and horizontal lines and figure (4) illustrates the direction of edge points.

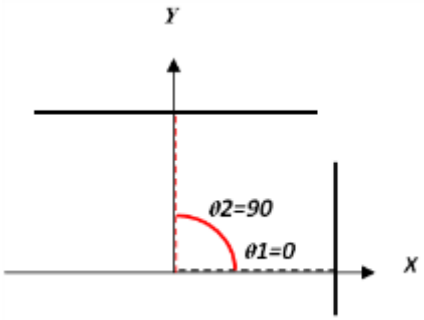


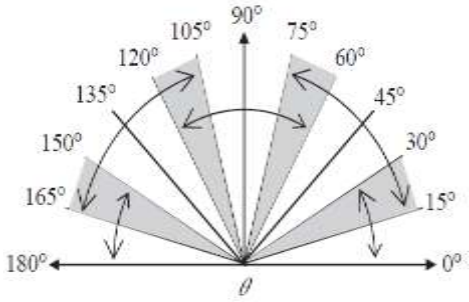
Figure (3), Representation of Vertical and Horizontal lines

The IHT matrix and IH, is $n\rho * n\theta$ where

A horizontal line will have θ=0 and ρ equal to the intercept with y-axis.

A vertical line will have θ=90 and ρ equal to the intercept with x-axis.

$$\text{angle range } \theta = \begin{cases} -90 & line\ direction = vertical \\ 90 & line\ direction = vertical \\ 0 & line\ direction = horizontal \\ 180 & line\ direction = horizontal \\ Otherwise & line\ direction = diagonal \end{cases}$$



Figure(4),Direction of edge point

Figure (5), shows the main steps for detecting vertical and horizontal line in IHT for input document image.

Rows and columns of IHT parameter space corresponded to ρ and θ values respectively. The elements in the IHT represent accumulator cells. Initially, the value in each cell is set to zero. Then, for every foreground pixels in image, ρwas calculated for every θ. ρ is rounded to the nearest row in IHT and increment the accumulator cell by one. At the end of this algorithm, a value of *Q* points in IHT space *IHT(ρ, θ)* means that *Q* points in the *x-y* plane lie on the line specified by θand ρ. Peak values represent possible lines in the input image.

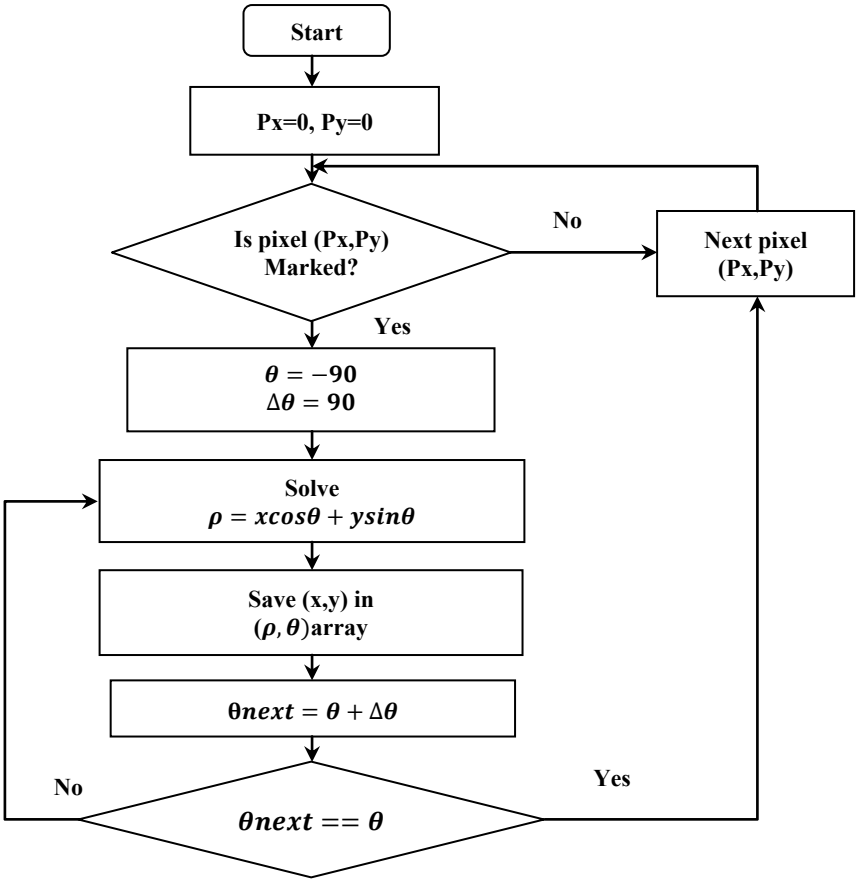


Figure (5), Flowchart for proposed detection vertical and horizontal lines method

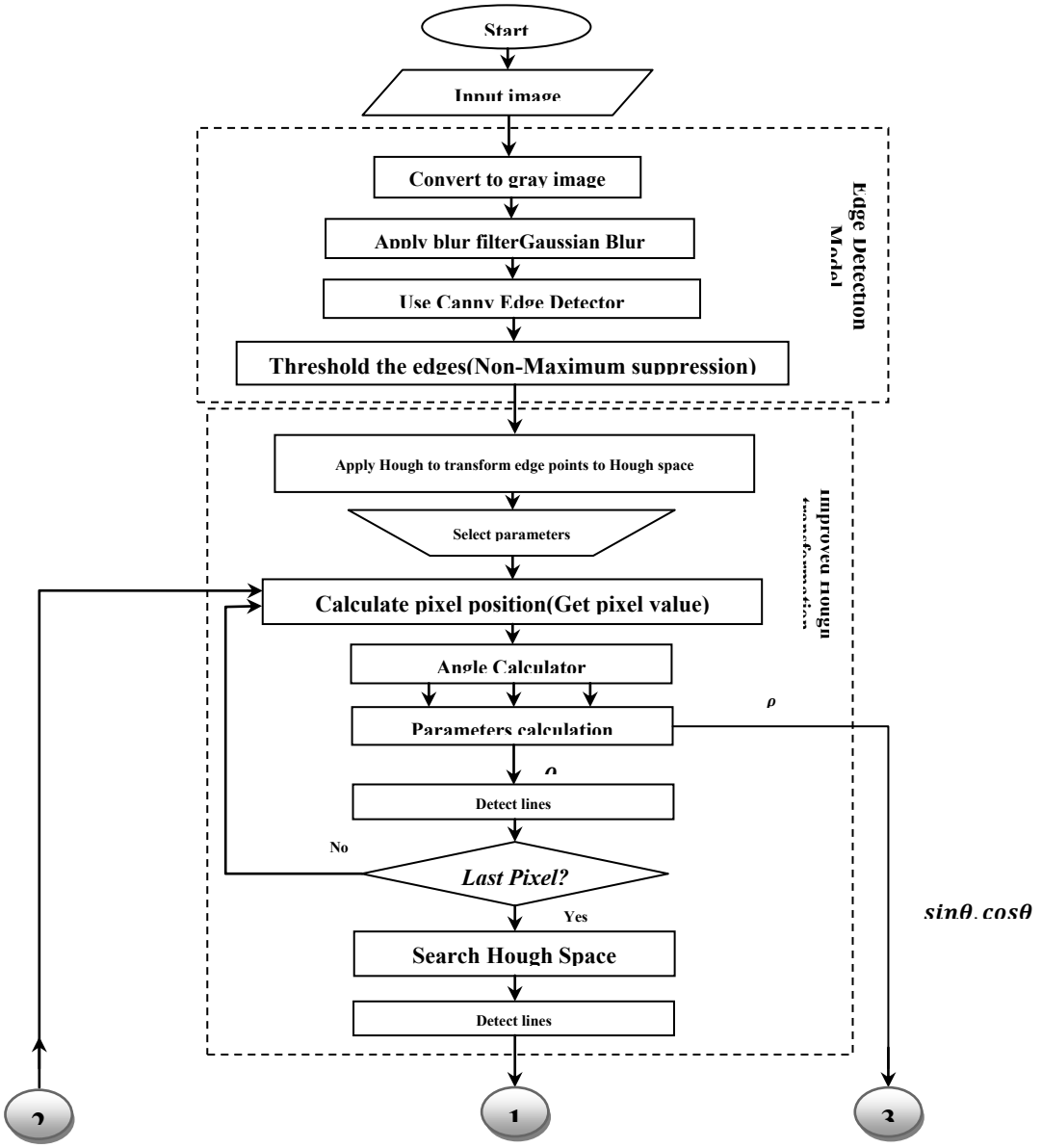
Proposed Approach for Detection and Extracting Tables

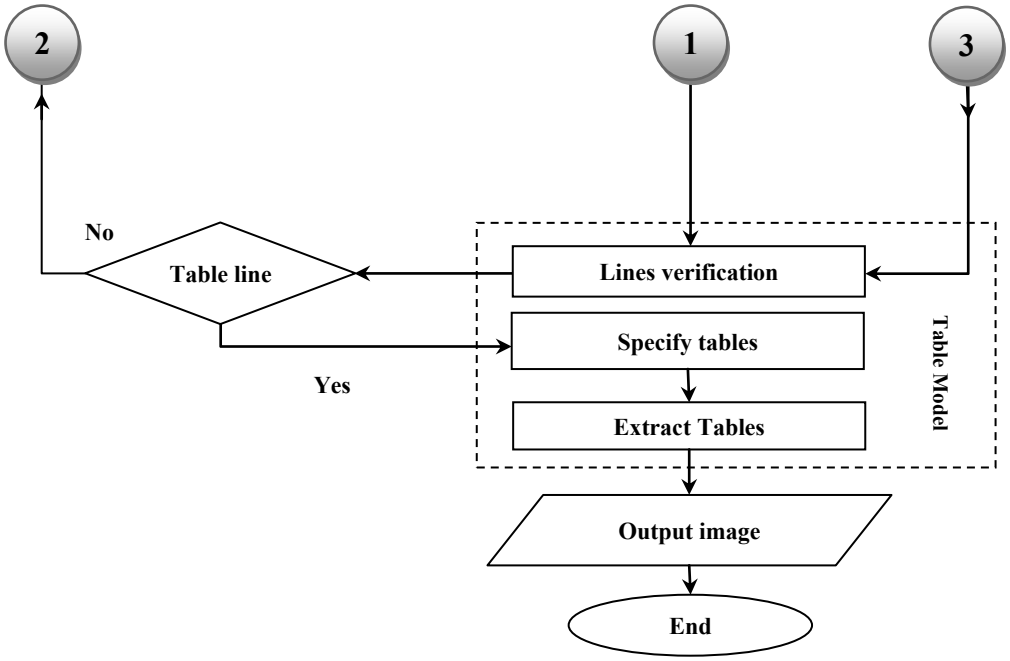
The proposed approach of table detection and extraction by using the improved Hough transform is illustrated through a flowchart of figure (6) that describes the detailed overall system flow. In addition, the detailed processed steps of the proposed system are represented in the algorithm (1) bellow, which lists the main steps of the proposed table detection and extraction system.

Each prominent edge points are voted and when all these points voted, The computation will stops,and the rest points being removed as supporting confirmation for the detected table's line and constraints is given, e.g. in the form of minimum line length a stopping rule can be tested before selecting a point for voting.

After detect the vertical and horizontal line, we start with table detection. Our table detection technique involves two distinct steps:

1. Check angle and Detection of line intersections and
2. Table detection - reconstruction.





Figure(6),Detailed Flowchart of the Proposed Algorithm

Here is an outline of the proposed algorithm of table detection and extraction using IHT:

1. Input image.
2. Output Extracted tables.
3. Specify width and height of the image and coordinates of the centre of the image to initialize the Hough array.
4. Apply Gaussian blur and edge detector on image to detect prominent edge and adjust parameters such that it will show all the edge in the image.
5. Apply a threshold to the edge image to clearly decide for each image point if it's an edge point or not.
6. Select prominent edge pixels from edge image and then vote it in array (Now find points and update array (Hough array)).
7. Update the accumulator with the selected pixel.
8. Specify the neighborhood size in which to look for the max peak.
9. Specify the number of discrete values of theta that should we check.
10. Calculate the maximum height the Hough array needs to have

hough_Height := (int) (Math.sqrt(2) * Math.max(height, width)) / 2;

11. Remove pixel from the input image.
12. Count the number of points

num_Points = 0;

13. Accumulate sin values and cosvaluess

sin_Cache:= new double[max_Theta];
cos_Cache:= sin_Cache.clone();
For (int t = 0; t <max_Theta; t++)
Begin
12.1. doublerealTheta = t * theta_Step;
12.2. sin_Cache[t] := Math.sin(real_Theta);
12.3. cos_Cache[t] := Math.cos(real_Theta);
End

14. Check if the value of peak in accumulator that it modified by recent pixel is higher than threshold *thr*. If not, then GOTO 5.
15. Validate the peaks to see if it belongs to valid lines, rather than noise.
16. Seek the neighbor of the peak in the accumulator to find the longest segment of pixels either continuous or contain a gap not greater than a given threshold.
17. Eliminate from the input image the pixels in the segment.
18. Initialize the lines vector that will return

// Continue if the array of Hough not empty.
// only take points that it is higher than threshold.
// seeking the local peaks higher than threshold to be draw
// Compute theta true value.
// Put the line on the lines-vector

19. Deselect from the accumulator all the pixels of the line that have previously voted.
20. Extracts lines.
21. Returns lines as a Vector.
22. If the line is longer than minimum length add this line into the output list.If not then GOTO 24.
23. Specify start and end point of each line.
24. Construct table using segmented lines,Convert the lines detected in Hough space back to image space to make them visible.
25. GOTO 4.
26. End.

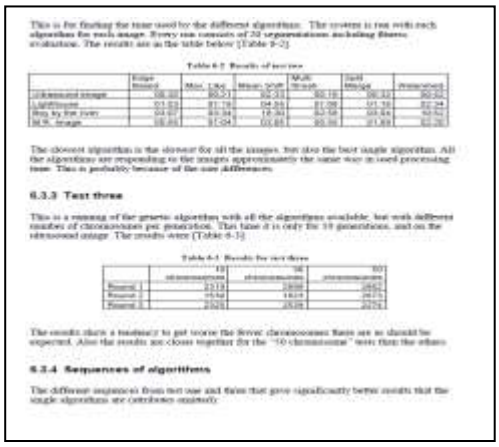
The Experimental Results

The implementation of the proposed IHT approach using different input images with different size and resolution to demonstrate the algorithm can detect and extract tables with high efficiency , and take the average for 10 runof our algorithm as evaluation time that is carried out using a 2.7 GHz CORE i7 processor.

The impact of the proposed method is verified with scanned images, three samples are shown in Figure6 (A, B and C).

A

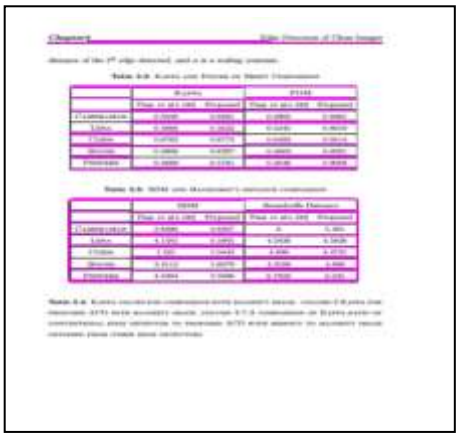
B



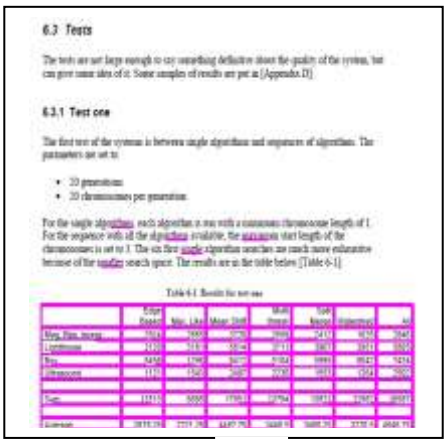
C

Figure (7),Samples of Document Image A, B and C

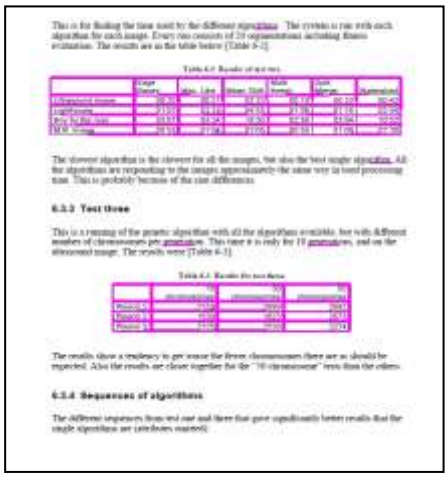
Figure (8) show the initial result for input images.



A



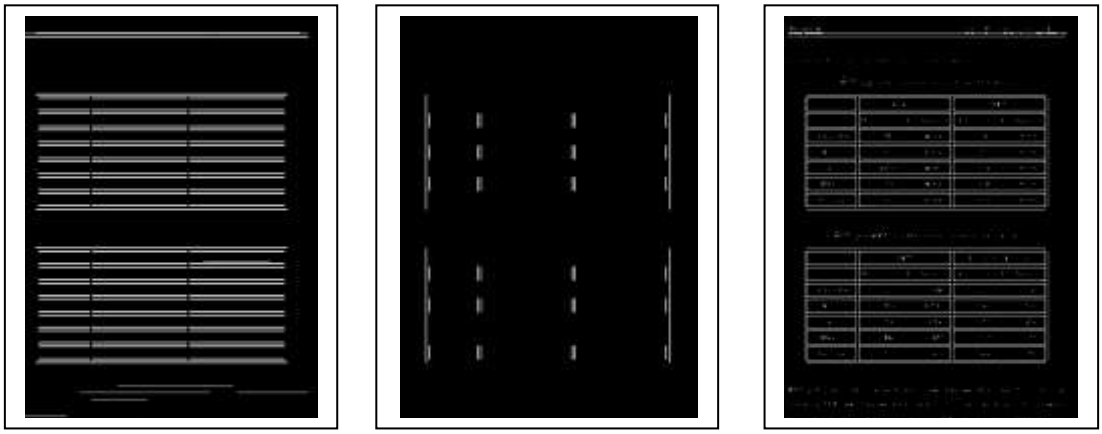
B



C

Figure (8), A, B and C Detecting Tables

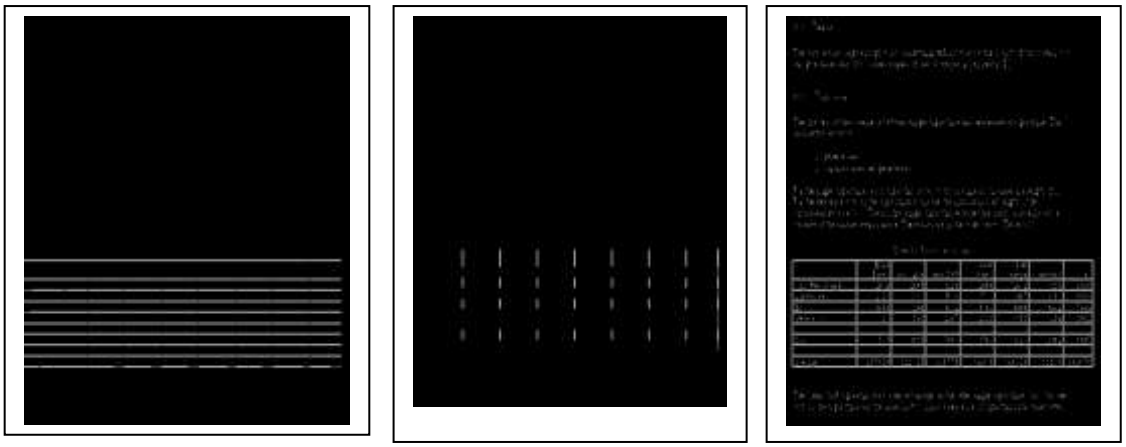
Figure (9), bellow shows the detected vertical and horizontal lines based on the IHT method for table detecting lines for input image A.



1 **2** **3**

Figure (9), (1) Detecting Horizontal line (2) Detecting Vertical Line (3) Combine the *V* and *H* lines, for input image A.

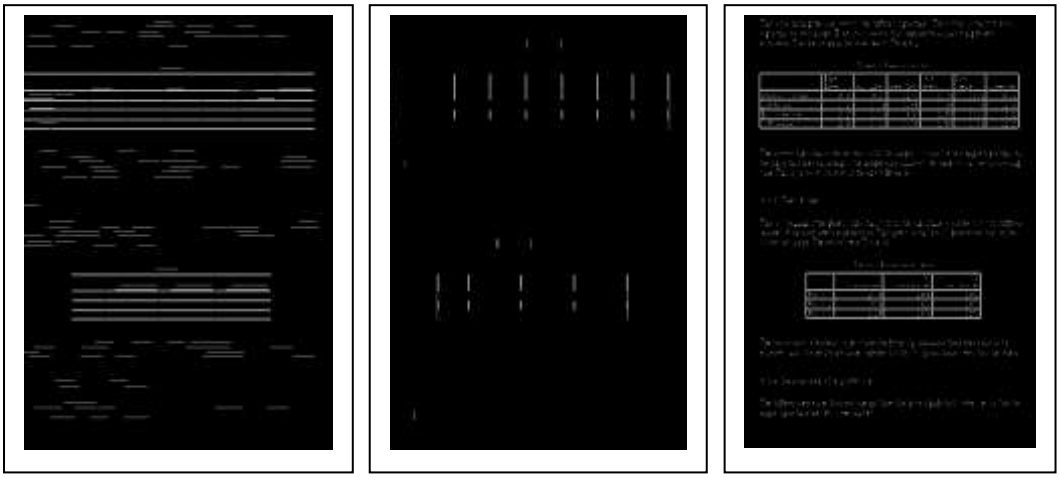
Figure (10), bellow shows the detected vertical and horizontal lines based on the IHT method for table detecting lines for input image B.



1 **2** **3**

Figure (10), (1) Detecting Horizontal line (2) Detecting Vertical Line (3) Combine the *V* and *H* lines, for input image B

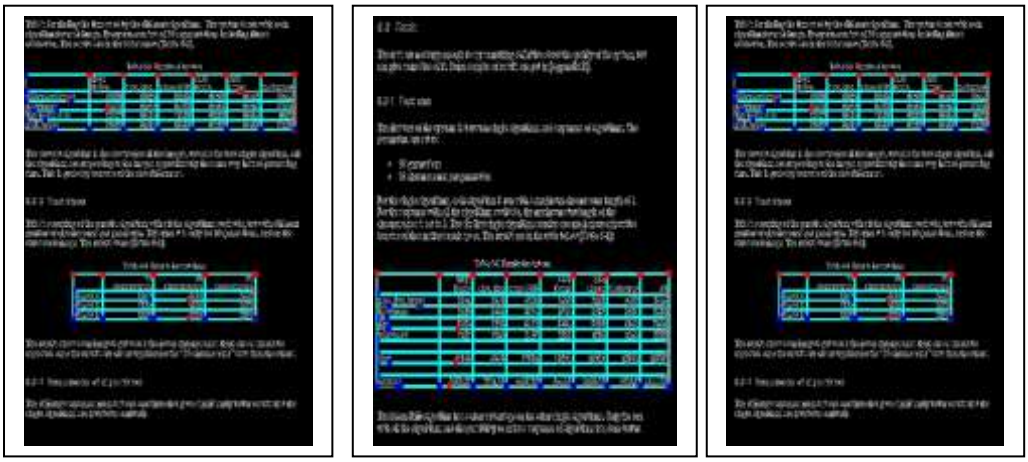
Figure (11), bellow shows the detected vertical and horizontal lines based on the IH method for table detecting lines for input image C.



1 **2** **3**

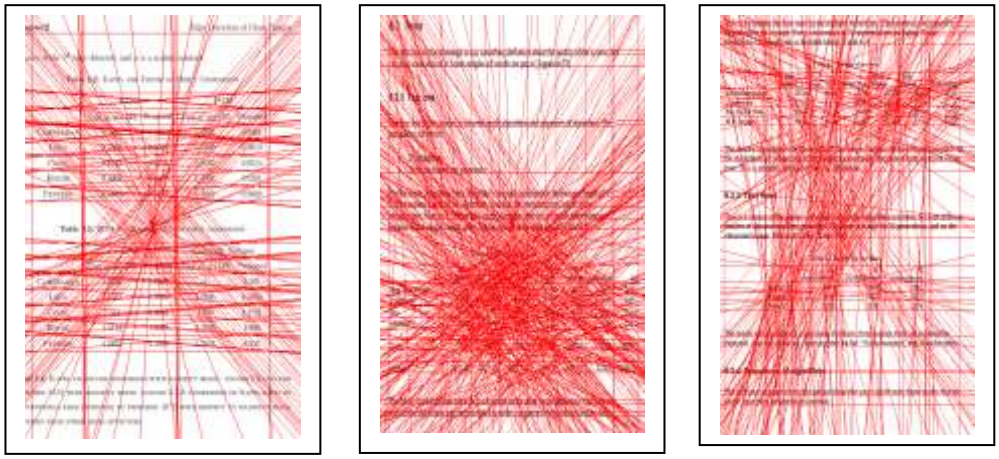
Figure 11: (1) Detecting Horizontal line (2) Detecting Vertical Line (3) Combine the *V* and *H* lines, for input image C.

In comparing the output of the proposed IHT and standard Hough Transform for table detection and extraction, figure (12), bellow shows the extracted tables in input images A, B and C using the proposed method, and figure (13) shows the extracted tables in input images A, B and C using the standard Hough Transform.
Also figure (14), represents the Hough space for each input image A, B and C.



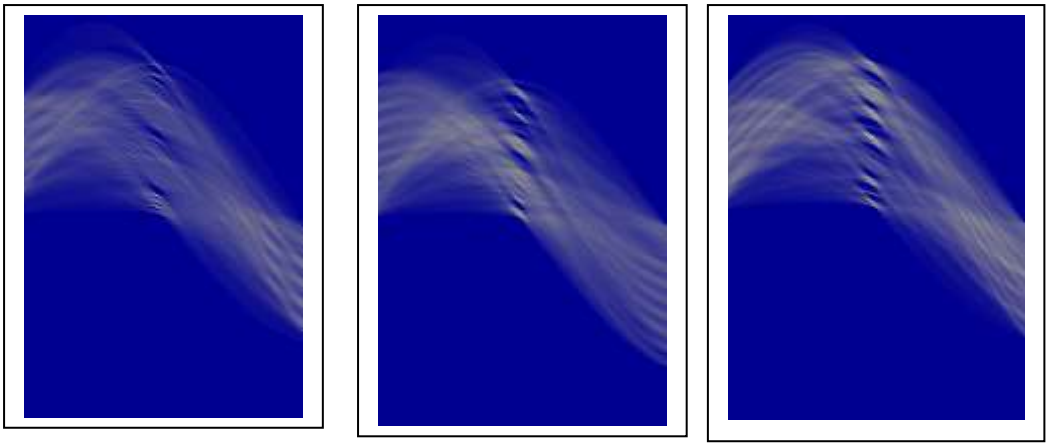
1 **2** **3**

Figure (12),1, 2 and 3 show extracted table for input images A, B and C



1 **2** **3**

Figure (13), 1, 2, and 3 Hough Lines of input images using standard Hough transform



1 **2** **3**

Figure (14), 1, 2 and 3 Hough Space of input images

Using dataset contains 95 images that are divided into four datasets, dataset 1, 2 contain a different number of tables, dataset 3 contain images not have tables and dataset 4 contain contaminated images (corrupted). Table (1) shows the accuracy of proposed IHT. From this table we can see the excellent result of dataset 1, 2, 3 and for dataset 4 the accuracy is low because the foreground pixel of table lines overlapping with text pixel or (and) background pixels (Lines of table inconspicuous).
Dependant on the following criterion to compute the accuracy of IHT:

$$Detected\_accurcy = \frac{No.\ of\ detected\ table}{Total\ No\ of\ table} * 100$$

Table (1), accuracy of detect tables in dataset

Dataset	Images	No. of Tables	No. of detected tables	No. of mis-detected tables	Accuracy
1	40	75	75	0	100%
2	10	15	15	0	100%
3	15	0	0	0	100%
4	30	40	29	11	72.5%

And when compare the execution results of proposed method with 3 input sample have the same size and resolution but with different number of tables, gained that the execution time to extract tables from images that contain a small number of tables much less than the images that contain many tables and obtain excellent accuracy for recognition all table lines as it is shown in Table (2). The proposed method gives a good performance on detection behavior.

Table (2), No. of detect lines and CPU time for 3 samples

Sample No.	Table NO.	No. of vertical line	No. of horizontal line	No. of detected line	time to extract tables (ms)
1	1	15	12	27	157
	2	35	32	67	
2	1	14	14	28	95
3	1	15	12	27	186
	2	35	32	67	
	3	30	28	58	

Moreover, the calculation of the recognition rate and the recognition precision for vertical and horizontal lines detection of the proposed method using IHT and the results are compared with the standard Hough transform (SHT) and General Hough transform (GHT).
The comparison method used is base on counting the number of detected line for each table in document image using SHT, GHT and IHT algorithm. In addition, we tested (implementation)SHT and GHT then compare them with IHT according to execution time.The number of detected lines As well as the time taken is listed in Table (3),when comparing the obtained result from IHT with SHT and GHT it can be seen that the IHT method gives the best results in terms of execution time and the number of discovered lines for tables in document image because IHT just detect the prominent vertical and horizontal line in image rather than detected all lines in all direction of image that the SHT and GHT are achieved.

Table (3), No. of recognized lines and CPU time for test document

	SHT		GHT		Proposed IHT		
	No. of Detected Line	CPU time *Ms*	No. of Detected Line	CPU time *Ms*	No. of Detected Line	CPU time *Ms*	Image dimension
Image A	142	341	232	290	63	152	492*615
Image B	320	280	360	210	147	142	595*578
Image C	233	398	278	340	44	158	569*705

CONCLUSIONS

This work improves the HT in order to increase the recognition rates of the tables in the document image by determining the length of line and the angle of the line. In fact, the IHT can detect lines with high accuracy. Besides that, it specifies their position in the target image. The recognition steps show that the IHT algorithm is adapted to recognize tables. Several tests have been conducted to show the efficiency of the method in the extraction of the table.

Trial results illustrate that IHT method can efficiently detect lines within tables in all tested sample, and show good performance when applied to scanned document image. TheIHT minimizes the total of processing needed to detect lines by exploiting the value of votes needed to reliably detect lines as well as reducing the execution time for table detection.

The IHT method gives better results than Standard Hough Transform in terms of its accuracy and time required for the calculation. On condition of extracting lines from a table, the priority of the threshold set for seeking peaks in the proposed method can save CPUtime if it compares with the traditional Hough transform methods (SHT and GHT).

REFERENCES

[1] R. Smith, "Hybrid Page Layout Analysis ViaTab-stop Detection", In Proc. Inter. Conference. on Document Analysis and Recognition, pages 241,Barcelona, Spain, July 2009.

[2] Marinai, Simone, Fujisawa and Hiromichi ,"Machine Learning in Document Analysis and Recognition", ISBN 978-3-540-76279-9 2008.

[3] Frédéric Bapst and Rolf Ingold,"Using Typography in Document Image Analysis ", Part I: RIDT'98 Recognition And Models Electronic Publishing, Artistic Imaging, and Digital Typography Volume 1375 of the series Lecture Notes in Computer Science pp 240-251, 22 May 2006.

[4] TanushreeDhiran andRakeshSharma,"Table Detection and Extraction from Image Document", International Journal of Computer & Organization Trends –Volume 3 Issue 7 – August 2013

[5] Cesari, F., Marinai, S., Sarti, L. and Soda, G.,"Trainable Table Location in Document Images", Proc. of the International Conference of Pattern Recognition, vol. 3, 236-240, 2002.

[6] Faisal Shafait and Ray Smith, "Table Detection in Heterogeneous Documents", Proceeding DAS '10 Proceedings of the 9th IAPR International Workshop on Document Analysis Systems Pages 65-72, 2010.

[7] Christopher Clark and SantoshDivvala, "Looking Beyond Text: Extracting Figures, Tables and Captions from Computer Science Papers", Papers from the AAAI Workshop, 2015.

[8] Zanibbi, R., Blostein, D., and Cordy,"A Survey of Table Recognition: Models, Observations, Transformations, and Inferences", In ICDAR, Volume 7 Issue 1, Pages 1 - 16, March 2004.

[9] B. Gatos, D. Danatsas, I. Pratikakis and S. J. Perantonis, ”Table Detection in Document Images”, National Center for Scientific Research “Demokritos”GR 15310 Athens, Greece, 2005.
[10] Gatos, B., Pratikakis, I. and PerantonisS.J.,"An adaptive binarization technique for low quality historical documents", IARP Workshop on Document Analysis Systems (DAS2004), Lecture Notes in Computer Science (3163), 102-113, 2004.
[11] " Canny edge detector ", https://en.wikipedia.org/wiki/Canny_edge_detector.
[12] Walaa M. Khalaf, Mohammed Ali Tawfeeqand Kadhum Al-Majdi, "Edge Detection Using Scaled Conjugate Gradient Algorithm in Back Propagation Neural Network", Eng. & Tech. Journal,Vol.32,Part (A), No.2, 2014.
[13] J. Illingworth and J. Kittler, "A survey of the hough transform. Computer Vision Graphics and Image Processing", Journal Computer Vision, Graphics, and Image Processing, Volume 44, Issue 1, pp: 87– 116, 1988.
[14] H. H. Abbas and Hussein Ali Hussein, "Line Detection Using Radon Transform", Eng.& Tech. Journal, Vol.28, No.6, 2010.