Use of Radon Transform in Orientation Estimation of Printed Text

Dr. AbdulSattar M. Khidhir
Mosul Technical Institute
Mosul, IRAQ
abdulsattarmk2@yahoo.com

***Abstract*—Alignment of images of printed text is needed as a preprocessing for the purpose of OCR. The test for an image alignment is easily developed by summation of pixels horizontally. However, the angel of orientation of the text is not easily estimated. In this paper, a method for estimation of this angle is presented. The Radon transform is used. The spaces between lines are considered as thick lines, which appear in Radon transform 3D plot as a hill (when background is white). Different angles were tested, which showed an acceptable accuracy of tilt angle measurement.**

Keywords: Radon Transform; Hough Transform; OCR; Text orientation.

I. INTRODUCTION

The Radon transform is named after J. Radon who showed how to describe a function in terms of its (integral) projections [1]. The mapping from the function onto the projections is the Radon transform [2].

Radon transform has found many various applications in science and engineering. It has the useful property of converting the information contained in a 2-D image into a series of 1-D projections. Many operations can be performed faster on the 1-D data set than on the equivalent 2-D image. This avoids the computational bottleneck that limits the speed of conventional 2-D digital processing systems.

It has recently been proposed that the transform can be used as the basis of the method for rapid feature extraction from image. The Radon transform can be obtained using the parallel processing capability of incoherent or coherent optical system. The 1-D data set can then be rapidly processed using dedicated hardware. Various image features that are invariant to object rotation and translation, can be calculated, including parameters based on the convex hull and polar projections [3].

In recent years the Hough transform and the related Radon transform have received much attention [4]. These two transforms are able to transform 2-D images with lines into a domain of possible line parameters, where each line in the image will give a peak positioned at the corresponding line parameters. This has led to many line detection applications within image processing, computer vision, and seismic [5].

The Hough Transform is a common tool for line detection. Since its introduction, much effort has been made to improve and understand it better. A comprehensive method for detecting straight line segments in any digital image is accurately controlling both false positive and false negative detections [6].

A new technique for rotation invariant texture analysis using Radon and wavelet transformation has been introduced by some authors [7]. Using this technique, the principal direction of the texture is estimated using Radon transform and then the image is rotated to place the principal direction at 0 degree. However, wavelet transform should then be employed to extract the features. For example, some textures may have straight lines along several directions. This may create ambiguity for the direction estimation. In that case, more complex methods may be employed to estimate the direction.

Radon transform-based linear feature detection has many advantages over other approaches such as its ability to detect line width and robustness in noisy images. It was used for centerline detection and line width estimation which are important for many computer vision applications, e.g., road network extraction from high resolution remotely sensed imagery. Several key issues that affect the centerline detection using the radon transform were investigated in some literature [8].

Radon Transform has been used in many other applications such as extracting the power line from aerial images acquired by an aerial digital camera onboard a helicopter [9].

II. THEORY

The idea behind the Radon transform is to integrate a function along a specific line and project the value of the integration in the Radon transform plane according to the distance of the line from the origin and its slope. To construct the Radon transform plane, the original function or shape is to be integrated along all possible distances and slop of lines in that plane and each integration along single line is projected as a point in the transfor plane.

Accordingly, if the shape contains a specific lin then the integration will have the maximum valı when it is taken along the line itself. Any oth integration path will have lower value. This fact used to detect straight lines in an image.

Mathematically the Radon transform is defin as:

$$R(\rho,\tau) = \int_{-\infty}^{\infty} f(x, \tau+\rho x)\, dx \quad (1)$$

Or

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\, \delta[y-(\tau+\rho x)]\, dy\, dx$$

Where ρ is the slope of the line along which the integration is taken and τ is the intersection with the y axis. See Fig.(1).

III. PRINTED TEXT ORIENTATION

In most OCR software the orientation of text in the scanner should be done manually. It is an easy task for the software to check if the text is oriented in horizontal way or not. This task is achieved by summing up the pixels horizontally.

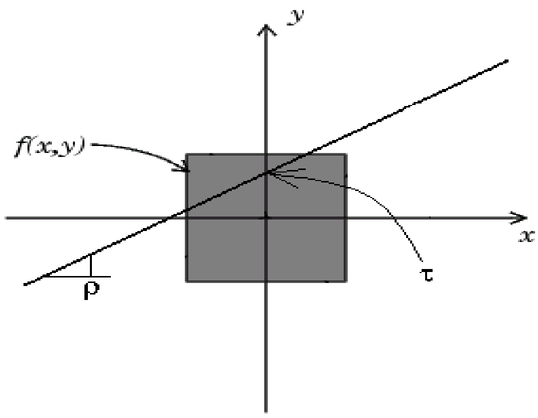
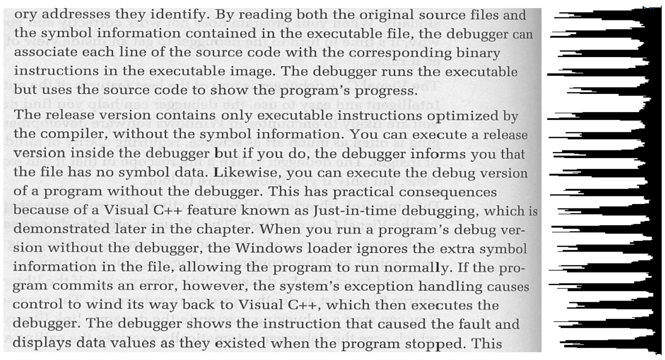


Fig.(1) Path for Integration for finding Radon Transform

The gap between lines should show zero value (or near zero value in case of noise) for the summation of pixels. Those minima are mostly periodical as shown in Fig.(2)..


ory addresses they identify. By reading both the original source files and the symbol information contained in the executable file, the debugger can associate each line of the source code with the corresponding binary instructions in the executable image. The debugger runs the executable but uses the source code to show the program's progress.

The release version contains only executable instructions optimized by the compiler, without the symbol information. You can execute a release version inside the debugger but if you do, the debugger informs you that the file has no symbol data. Likewise, you can execute the debug version of a program without the debugger. This has practical consequences because of a Visual C++ feature known as Just-in-time debugging, which is demonstrated later in the chapter. When you run a program's debug version without the debugger, the Windows loader ignores the extra symbol information in the file, allowing the program to run normally. If the program commits an error, however, the system's exception handling causes control to wind its way back to Visual C++, which then executes the debugger. The debugger shows the instruction that caused the fault and displays data values as they existed when the program stopped. This

Fig.(2) Summation of pixels of scanned image horizontally

However, if the text is disoriented, then then curve resulting from summing up the pixels horizontally will not give gaps between lines, which is needed in the step of character recognition from extracted line. See Fig.(3).

To achieve the task of finding the angle of orientation one could tilt the image in steps of small angles until a form of curves similar to Fig.(2) appears. However, this is a time consuming task and consumes a lot of computational power.

In this work, the idea behind the Radon transform is used to find the angle of orientation in one step. This is done by taking Radon transform of the disoriented image. In this case the text lines and

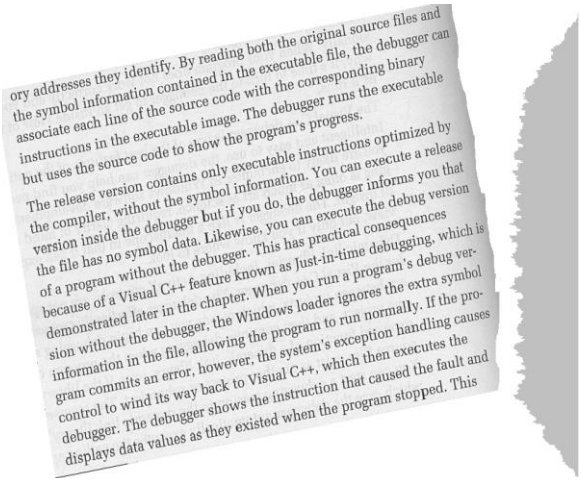


Fig.(3) Horizontal summation of pixels for a tilted image.

the spacing between lines will form straight lines which will appear in Radon transform as points. And since all text lines are parallel to each other (all have the same slope) then each line will appear in Radon transform plane as a point; all with the same ρ but with different τ. This could be noticed in Fig.(4).

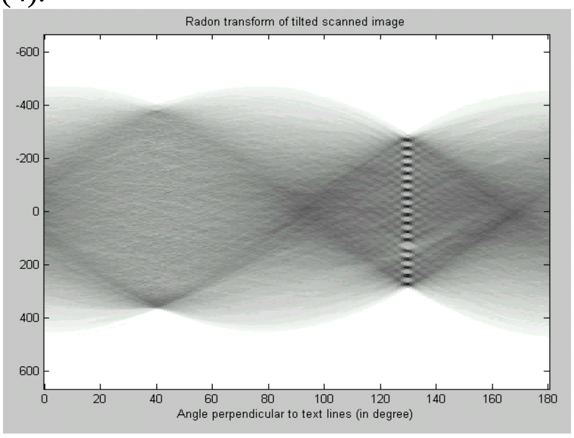


Fig.(4) Typical Radon transform of a scanned text image (Inverted in colors)

IV. EXPERIMENTAL WORK

In this work a sample of text has been treated using Radon transform. The computation was performed using MATLAB which includes a built-in Radon transform. In MATLAB, Radon transform is performed with respect to the angle of perpendicular line instead of the slope.

Experiments where repeated for various tilting angles (0^0 10^0 20^0 30^0 40^0 & 50^0) of the original image. Each resulting Radon transforms had been summed vertically to find the peak. This peak represents the angle of orientation of the original image. These experiments where repeated for three different images and the results are shown in Fig.(5a..5f, 6a-6f). The angles measured are the position of the peak of the curve which results from summing up the Radon Transform image vertically.

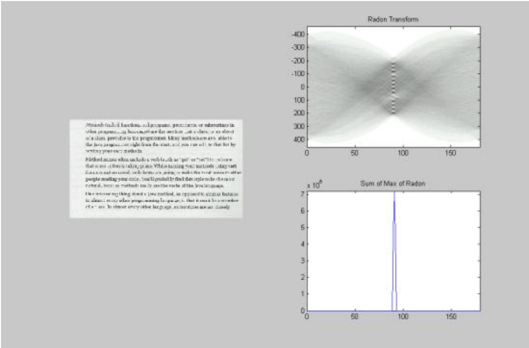
As could be noticed from the results, the accuracy of orientation is good. Table (1) compares the results when different angles were used. It is noticed that when the image is noisy (Fig. 6a-6f), then the two sides around the center peak show different levels of fluctuations. However, the peak value position still gives a result of enough accuracy.

TABLE I. Comparison of Tilt Angles of Text with those Resulting from Radon Transform

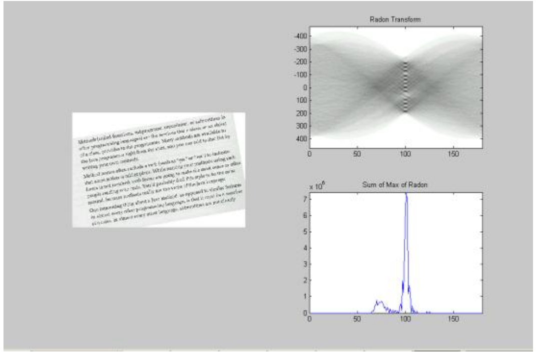
The Figure	Actual angle (^{0})	Measured	Absolute error
5a	0	359	-1
5b	10	11	1
5c	20	20	0
5d	30	30	0
5e	40	41	1
5f	50	51	1
6a	0	1	1
6b	10	12	2
6c	20	21	1
6d	30	31	1
6e	40	39	-1
6f	50	52	2

V. CONCLUSION

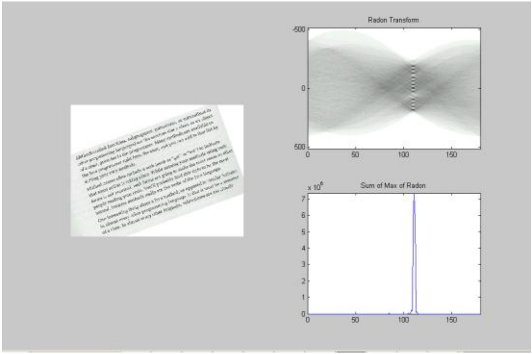
The use of Radon Transform for finding the tilt angle of printed text has been shown. From the results presented it is clear that Radon Transform is an effective method for finding such tilt angle. When such an angle is found, any optical character recognition software may be used. The results of this work are not associated to a specific language or character set. Although the tests were performed on a printed text, However, a handwritten text may show similar results if it is neatly written and the spaces between lines are uniform. Although the proposed method for principal direction estimation is suitable for most of the ordinary textures, more complex textures may need more complex techniques.



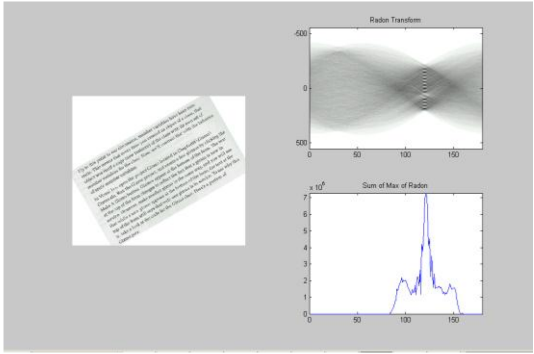
(a)



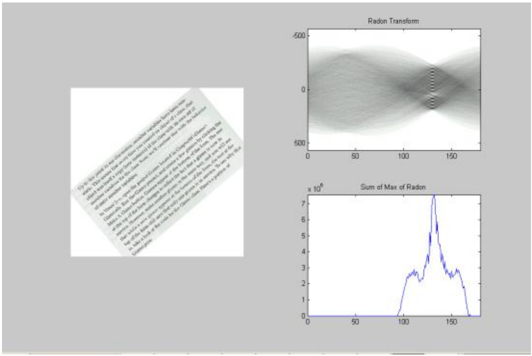
(b)



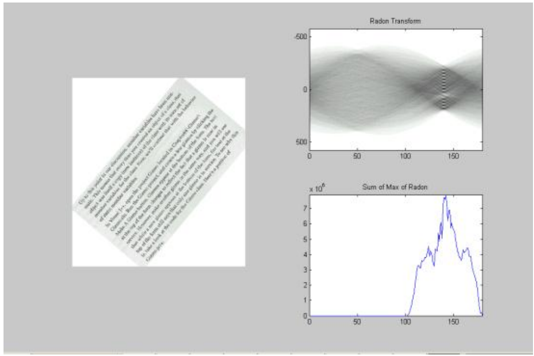
(c)



(d)



(e)



(f)

Fig. (5) Samples of Processing of non-Noisy Image.

REFERENCES

[1] J. Radon, *Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten. Berichte Sächsische* Akademie der Wissenschaften, Leipzig, Mathematisch-Physikalische Klasse, 69:262–277, 1917.

[2] M. van Ginkel, C.L. Luengo Hendriks and L.J. van Vliet *A short introduction to the Radon and Hough transforms and how they relate to each other* Number QI-2004-01 in the Quantitative Imaging Group Technical Report Series, 2004.

[3] D. Nguyen, *Determination of Translation and Rotation of an Object from its Radon Transform via Neural Network*, IEEE International Conference on Systems, Man, and Cybemetics, pp. 3281 -3284 vol. 5, 2001.

[4] P.V.C. Hough. *Method and means for recognizing complex patterns*. US patent nr. 3069654, 1962.

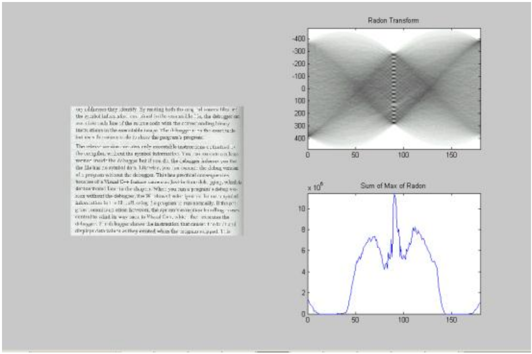
[5] P. Toft, *The Radon Transform - Theory and Implementation*, PhD Theis, IMM, DTU, 1996.

[6] R. G. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, *On straight line segment detection*, J. Math. Imaging Vis.,vol. 32, no. 3, pp. 313–347, 2008.

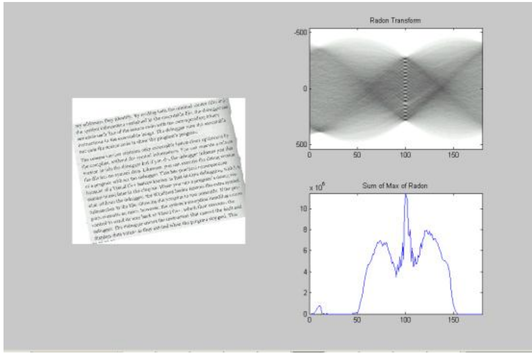
[7] K. Jafari-Khouzani and H. Soltanian-Zadeh, *Radon Transform Orientation Estimation for Rotation Invariant Texture Analysis*, IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol. 27, No. 6, June 2005

[8] Q. Zhang and I. Couloigner, *Accurate Centerline Detection and Line Width Estimation of Thick Lines Using the Radon Transform,* IEEE Trans. On Image Processing, Vol. 16, No. 2, Feb. 2007

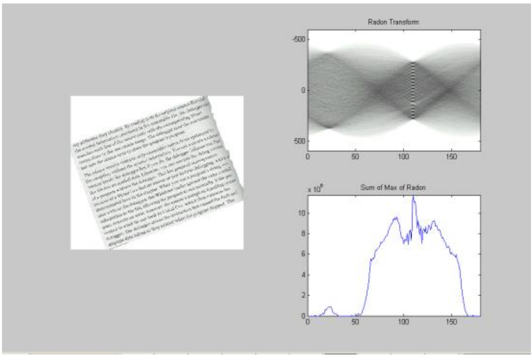
[9] G. Yan, C. Li, G. Zhou, W. Zhang, and X. Li, *Automatic Extraction of Power Lines From Aerial Images*, IEEE Geoscience and Remote Sensing Letters, Vol. 4, No. 3, July 2007



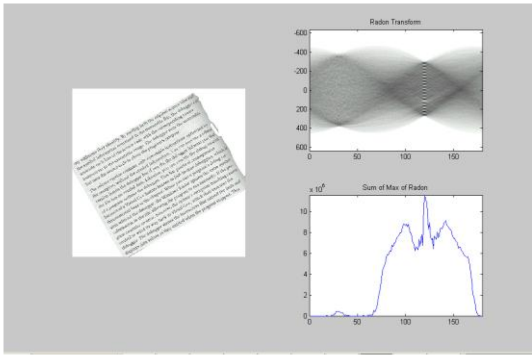
(a)



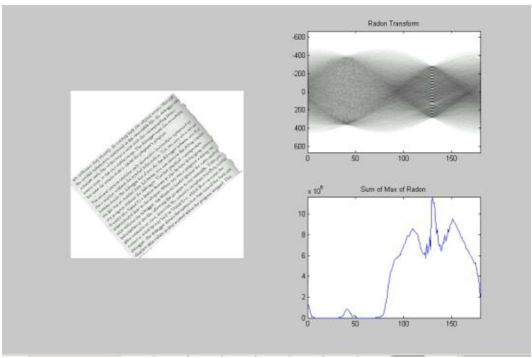
(b)



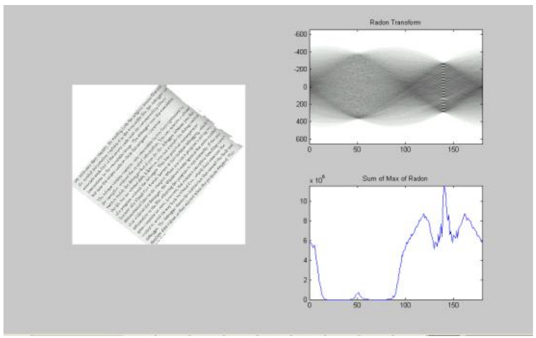
(c)



(d)



(e)



(f)

Fig. (6) Samples of Processing of Noisy Edge Image (the right edge of the text is darker and not straight)