Pattern Recognition 41 (2008) 3528 – 3546

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Hough transform based fast skew detection and accurate skew correction methods

Chandan Singh[a,*], Nitin Bhatia[b], Amandeep Kaur[c]

[a]Physical Sciences, Department of Computer Science, Punjabi University, Patiala-147002, India
[b]Department of Computer Science, DAV College, Jalandhar-144008, India
[c]Department of Computer Science, Punjabi University, Patiala-147002, India

ARTICLE INFO

Article history:
Received 3 January 2007
Received in revised form 29 May 2008
Accepted 5 June 2008

Keywords:
Skew detection
Skew correction
Hough transform
Rotation

ABSTRACT

The Hough transform provides a robust technique for skew detection in document images, but suffers from high time complexity which becomes prohibitive for detecting skew in large documents. Analysis of time complexity on various stages of skew detection process is carried out in this paper. A complete skew detection and correction process is divided into three parts: a preprocessing stage using a simplified form of block adjacency graph (BAG), voting process using the Hough transform and de-skewing the image using rotation. Skew correction phase, which is hitherto a neglected area, is analysed for the quality of de-skewed images with respect to the type of rotation. Fast algorithms for all the three stages are presented and exhaustive analysis on time complexity is conducted. It is shown that the overall time taken for the whole process is less than one second even for very large documents. It is also observed that time taken in rotation is as significant as in skew detection which is reduced with the help of fast algorithms using integer operations. While the BAG algorithm is found to be effective for documents with Roman script, it does not provide satisfactory results for Indian scripts where headline is a part of a script.

1. Introduction

Optical character recognition (OCR) is an important activity in office automation system which is now common in many offices because it facilitates a printed document to be automatically read by the computer. Of course, it has several other applications also including automatic reading of postal addresses, forms, government records, credit card imprints, postal mails, etc. An OCR system converts the image form of a document into a text in editable form. The editable text is in the form of character codes (ASCII or UNICODE) whereas the image is in bitmap form. While the primary goal of an OCR system is to convert an image into text, it also results in reduced space which is very useful for storage and transmission of documents. The process of conversion from image to text is a tedious task and involves several steps. The performance of an OCR system depends on the cumulative accuracy and speed of these steps. The steps include scanning, preprocessing, processing and post-processing. Skew detection is a part of preprocessing stage which affects the performance of an OCR system significantly if not detected and corrected properly.

It is not only the skew detection process which is important, its proper correction is also important. The authors have not found any analysis on the issue of skew correction which is normally perceived to be a trivial task by researchers. After the skew is detected, it is used to perform the rotation in the opposite direction to remove the skew. This procedure seems to be straightforward, but it is shown in this paper that it needs analysis of time complexity of rotation algorithms and the quality of de-skewed documents.

The methods of skew corrections fall broadly into three categories: Hough transform, projection profile and nearest neighbours. O'Gorman [1] has discussed a detailed account of these categories. Methods based on cross correlation and Fourier transform and some other hybrid methods are also available. Overview of these methods and a few recent hybrid techniques can be found in Refs. [2–6]. The Hough transform based technique was first used by Srihari and Govindaraju [7]. Modifications of the Hough transform approach are proposed by [8–10]. Despite its high time and space complexity, the methods based on the Hough transform are very popular because of the robustness and simplicity of the Hough transform [4,7]. Yu and Jain [2] discuss a multistage strategy to make the algorithm fast and more accurate. In their method, the amount of input data is reduced by considering the centroids of connected components instead of processing all image pixels in the first stage that is called the block adjacency graph (BAG). In the second and the third stages a hierarchical approach is used where initially a coarser Hough space is used to have a rough estimate of the skew and later a refinement is made to increase the accuracy. The method is, nevertheless, appealing as it reduces the number of image pixels significantly and reduces noise

* Corresponding author.
E-mail addresses: chandan.csp@gmail.com (C. Singh), n_bhatia78@yahoo.com (N. Bhatia), amandeep@pbi.ac.in (A. Kaur).

0031-3203/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2008.06.002

introduced due to black margins, lines in tables or forms, figures, etc. The noise reducing capability of the proposed methods works well for Roman script. It is shown in this paper that the method does not work well for scripts which have lines as a major part of the script. For example, many Indian scripts such as Devanagri, Bangla, Gurmukhi, etc. have lines, called headlines, connecting letters of a word. This special feature is very helpful in many operations including skew detection process because the number of votes along the direction of headline is much more as compared to any other direction. The BAG run, however, replaces the headline by a pixel and thus the important feature of these scripts turns into a major disadvantage when the BAG algorithm is applied. Jiang et al. [3] use the Hough transform to speed up the process by resorting to three steps. In the first step, document is divided into several blocks, each block consisting of 100 rows. In the second stage, "detection points" of each block are recorded which are used to estimate skew in coarse form. Finally, the accurate skew is obtained by choosing the peak in the Hough transform in the neigbourhood of the coarser skew. Manjunath et al. [6] have detected skew in two stages. In the first stage, selected characters from document images are "blocked" and thinning is performed over the "blocked" regions. In the second stage, the thinned coordinates are fed to the Hough transform [7]. The thinning process is also computation intensive and no time complexity has been provided by the authors on this account and, therefore, no proper conclusion can be made about the overall speed of the algorithm. Le et al. [9] use bottom pixels of the candidate objects within a selected region for the Hough transform which was later modified by Pal and Chaudhuri [10]. Yin [11] discusses a skew detection method which first smoothes the black runs and locates the black–white transition to emphasize the text lines. An improved Hough transform is then used to detect skew.

It is clear from the above discussion that the Hough transform forms a base for many skew detection techniques. We thus focus on the Hough transform whose efficiency is well established even under noisy conditions. It is restrictive in use primarily because of its slow speed and the requirement of large space. The space complexity can be minimized by modifying the algorithms and making them suitable for their implementation on PCs which are available now a day with high memory, 256 MB or above. Thus we focus our attention on the issue of speed and present algorithms which can detect skew to desirable accuracy, although an accuracy of 1° is sufficient for many applications using A4 size documents scanned at 300 dpi. We have performed exhaustive experiments on different documents collected from different sources and analysed skew detection in the documents varying in scripts, scanning resolutions, fonts and sizes. The second issue addressed in this paper pertains to skew correction. We have not come across any analysis on the skew correction which, if not performed properly, may affect the performance of an OCR system. One of the authors, Singh [12], has recently observed that quality of a rotated image is affected by the type of rotation. Two types of rotation exist: forward rotation and inverse rotation. Theoretically, both rotations are expected to provide the same result. This, of course, is true for the continuous functions but the results are different for the discrete functions, as shown in this paper.

The rest of the paper is organized as follows: The basic Hough transform with its major characteristics is presented in Section 2. The preprocessing stage using the modified BAG algorithm is discussed in Section 3. Fast algorithms for voting using the Hough transform are mentioned in Section 4. Section 5 discusses skew correction process through forward and inverse rotations. Quality of de-skewed document images as a function of the type of rotation is analysed. This section also presents fast algorithms for the rotation and their time complexities. Results of skew detection and correction and analysis of time taken by different phases for few document images is given in Section 6 followed by conclusions in Section 7.

2. The Hough transform

The Hough transform maps a line in spatial domain to a point in the Hough parametric space. The Hough parametric space is a 2-D space and is described in the form (a, b) in Cartesian coordinates system where the slope is a and b is the intercept of the line passing through the point (x_0, y_0). The transform is

$$b = -ax_0 + y_0 \tag{1}$$

Fig. 1 depicts the Hough transform in Cartesian system.

Since the dynamic range of both the parameters, a and b, is $[-\infty, \infty]$, it is restricted to finite domain by splitting the range of a in two parts : $-1 \leqslant a \leqslant 1$, and $-\infty \leqslant a < -1$ and $1 < a \leqslant \infty$, as given by following equations:

$$b = \begin{cases} -ax_0 + y_0, & -1 \leqslant a \leqslant 1 \\ -ay_0 + x_0 & \text{otherwise} \end{cases} \tag{2}$$

The parametric space $b = f(a)$, where $f(a)$ is the right-hand side of Eq. (2), is drawn in Fig. 2, for $-1 \leqslant a \leqslant 1$, for an image whose resolution is $H \times W$ where H is the image height (along vertical direction) and W is the image width (along horizontal direction). It is clear from Fig. 2 that, at $a = 0$, $b_{\min} = 0$ and $b_{\max} = H$, at $a = 1$, $b_{\min} = -W$ and $b_{\max} = H$, and at $a = -1$, $b_{\min} = 0$, $b_{\max} = W + H$. Thus in the region $-1 \leqslant a \leqslant 1$, b spans from $-W$ to $W + H$.

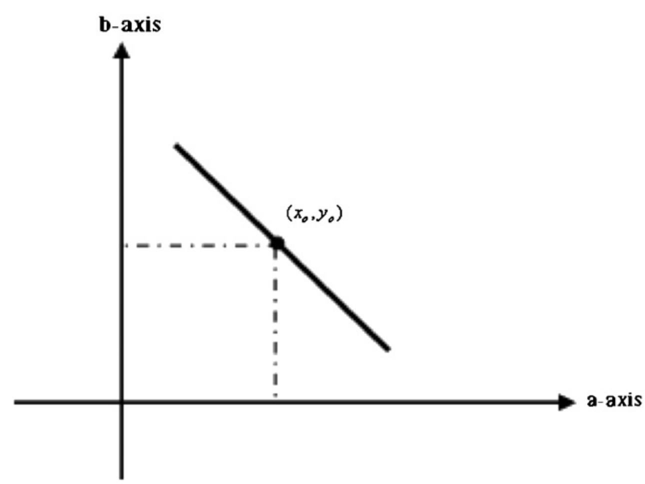

Fig. 1. The Hough transform in slope–intercept form for the line, $b = -ax_0 + y_0$.

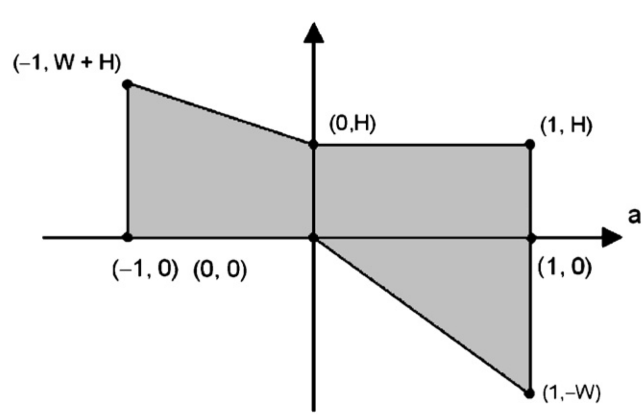

Fig. 2. Boundary of the Hough space for $-1 \leqslant a \leqslant 1$.