A SKEW DETECTION METHOD FOR 2D BAR CODE IMAGES BASED ON THE LEAST SQUARE METHOD

YING-HONG LIANG, ZHI-YAN WANG

School of computer science and engineering, South China University of Technology, GuangZhou 510640, China
E-MAIL: lyhwin@139.com, wzhyan@ieee.org

Abstract:

A robust and fast algorithm for skew detection in 2D bar code images is proposed in this paper. It is based on the least square method. Unlike the methods based on Hough transforms that are computationally expensive, it quickly obtains skew angles making it applicable to real-time applications. This method includes two processes, the segmenting process searches for the bar code region, and then the line fitting process fits the borderline and obtains the skew angle. Experimental results show this method reduces the running time.

Keywords:

Skew detection; 2D bar code; Least square method; Hough transform

1. Introduction

One dimension (1D) bar codes have been the most important automatic recognition symbol for almost eighty years and are widely used in all kinds of business, but they have some limitations. Recently various 2D bar codes have been developed. Compared to 1D bar codes, they can store more information and have an error-correcting capability. To use image processing to identify 2D bar codes, the first step is the preliminary processing. By converting the original 2D bar code to a binary image, the skew angle of its code area is calculated and the code area is segmented from the binary image.

Several methods have been developed for skew angle detection. The most common methods are based on the Hough transform, which consists of mapping points in Cartesian space (x, y) to sinusoidal curves in (ρ, θ) space via the transformation:

$$\rho = x\cos(\theta) + y\sin(\theta) \tag{1}$$

According to Stuart [1], the longest lines in a document image can be found through a Hough transform, and because these lines represent the textual region, the document skew angle can be calculated from their slopes. However, it is difficult to obtain an accurate angle through the basic Hough transform method owing to the diffuseness of the peak value. In addition, the Hough transform process is computationally expensive. Therefore the basic Hough transform is not applied to real-time applications.

Zhang [2] introduced a method for determining the skew angle of a bill image based on directional projection. By projecting the bill image at different angles, the method searches for the angle at which the projection width is the narrowest. The obtained angle is regarded as the slope angle of the bill image. This method is also time-consuming, so it cannot be used in real-time applications even though it can get accurate results.

Other methods for the detection of skew angles of document images are based on Fourier transforms and the nearest neighbor clustering method [6-7]. However, most of these methods are mainly designed for machine printed documents and are not suitable for 2D bar code images.

In this paper we introduce a method for the skew angle estimation of 2D bar code images based on the least square method (LSM), which is very fast and is able to work on high-noise 2D bar code images. This paper is organized as follows. In Section 2, the structure of the PDF417 bar code is introduced. Section 3 describes the skew detection algorithm. Experimental results are shown in Section 4 and conclusion appears in Section 5.

2. PDF417 bar codes

The Portable Data File (PDF417) bar codes were developed by Symbol Technologies in 1992 and since have been an international standard of 2D bar codes. The PDF417 bar code is a stack 2D bar code composed of multi-row 1D bar codes (see Figure 1). The code is comprised of the following patterns: left blank pattern (a) start pattern (b), left row indicator (c), codewords (d), right row indicator (e), stop pattern (f), and right blank pattern (g).

Figure 1 shows that the PDF417 bar code is made up of a collection of thick lines. Because the image has no

1-4244-0060-0/06/$20.00 ©2006 IEEE

obvious thin lines, it is hard to obtain an accurate skew angle through the Hough transform. Normally the bar code labels are stuck on other objects, such as bills and protective marks, thus the bar code images captured by video cameras or scanners have high noise, which includes the background noise (from the objects containing them), smudges, scratches and glare interference. The noise and the borderlines of these objects can muddle the detection results.

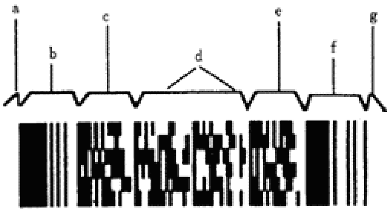


Figure 1. The structure of the PDF417 bar code

3. The skew detection approach

3.1. Image segmentation

Figure 2(a) shows the original bar code image captured by a video camera has background noise and the glare interference. The skew information of the bar code region is crucial to skew detection, but the noise makes the searching of eigen-points difficult. For the bar code images that have high contrast and have no noise, the eigen-points can be found easily in the original image but fail, when the images are low-contrasted and have high noise. To make lines fit precisely and quickly, the first step should be segmenting the original image and detecting the bar code region.

The segmentation process of the bar code region is based on the projection method. By projecting the original bar code image both in the horizontal and vertical direction, the x-coordinates and y-coordinates of the bar code region are found. The bar code region is the maximal low gray region in the original image, so it displays a wave crest which has the maximal width both in the horizontal and vertical projection images (see Figure 2(b) and 2(c)). The region needn't be very precise for borders.

The segmentation process consists of the following phases:

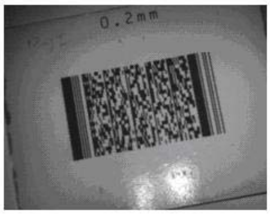
1. In order to reduce the interference of the background noise that causes the projection image to have many wave crests of noises, the original image is edge enhanced with 3x3 Sobel masks in the horizontal and vertical directions. [3]
2. Calculate the horizontal and vertical projection. For pixel (x, y) and its gray value G (x, y), the horizontal and vertical projection are defined as:

$$P_i(j) = \sum_i G(i, j),\ P_j(i) = \sum_j G(i, j) \qquad (2)$$

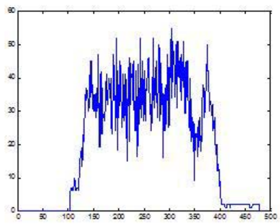
After selecting an appropriate threshold w, search for the longest continuous region in the horizontal projection where $P_i(j) > w$. The starting coordinate j_s and ending coordinate j_e of this region is the top and bottom coordinates of the bar code region. The left coordinate i_s and right coordinate i_e can be calculated by the same approach.

3. Clip the original image with the obtained coordinates and threshold the clipped image. Figure 2(d) is the resulting image.

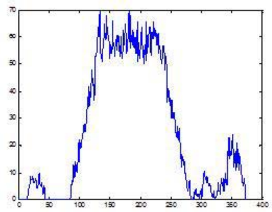
In contrast with the original image, Figure 2(d) is “clean” and the contrast between the bar code region and its background is enhanced. The bar code region can be segmented precisely from the original image by the projection method. The resulting image is very useful to the processing of line fitting and bar code decoding.



(a) The original bar code image



(b) The horizontal projection



(c) The vertical projection

(d) The segmented bar code region (binary image)

Figure 2. The segmentation of the bar code image

3.2. Searching for the eigen-points

The first step to use the LSM to fit a line is to determine the set of eigen-points. The four edge lines of the image are the choices. Figure 2(d) shows that the left and right borders are obvious and continuous, but the top and bottom borders are discontinuous. The algorithm of selecting the eigen-points is composed of three steps:

1. Check the continuity of the left and right borderlines. If one of the lines is continuous, use it to fit, otherwise

use the top line or the bottom line.

2. The first and the last black pixel of the line are eigen-points; next choose several black pixels from the middle as eigen-points.
3. Put all the coordinates of the selected points in an array. This array is the set of eigen-points to be fitted.

3.3. Calculation of the skew angle

The method of least squares assumes that the best-fit curve of a given type is the curve that has the minimal sum of the deviations squared (least square error) from a given set of data.

For a set of eigen-points in an image: $\{(x_i, y_i), i=0,1,\ldots,m\}$, the function $y = S^*(x)$ represents the fitting curve that comes closest to pass through all of the points. The fitting curve $S^*(x)$ has a deviation from each of the eigen-points:

$$\delta_i = S^*(x_i) - y_i, \quad i = 0,1,\ldots m, \quad \delta = (\delta_0, \delta_1, \ldots, \delta_m)^T \tag{3}$$

According to LSM, the best fitting curve has the property that:

$$\| \delta \| = \sum_{i=0}^{m} \delta_i^2 = \sum_{i=0}^{m} [S^*(x_i) - y_i]^2 = \text{a minimum} \tag{4}$$

Here we use a straight line $S^*(x)=kx+b$ to approximate the given set of eigen-points, where k and b are unknown coefficients.

For each eigen-point: the least square error is: $S^*(x)-y_i=kx_i+b-y_i$, and the sum of the squares of the errors is:

$$A(k,b) = \| \delta \| = \sum_{i=0}^{m} [kx_i + b - y_i]^2 \tag{5}$$

To make $A(k, b)$ = min, the unknown coefficients k and b must yield zero first derivatives.

$$\begin{cases} \dfrac{\partial A}{\partial b} = -2\sum_{i=0}^{m}(y_i - b - kx_i) = 0 \\ \dfrac{\partial A}{\partial k} = -2\sum_{i=0}^{m}(y_i - b - kx_i)x_i = 0 \end{cases} \tag{6}$$

The unknown coefficients k and b can therefore be obtained:

$$\begin{cases} b = \dfrac{\sum_{i=0}^{m} y_i \sum_{i=0}^{m} x_i^2 - \sum_{i=0}^{m} x_i \sum_{i=0}^{m} y_i x_i}{(m+1)\sum_{i=0}^{m} x_i^2 - (\sum_{i=0}^{m} x_i)^2} \\ k = \dfrac{(m+1)\sum_{i=0}^{m} y_i x_i - \sum_{i=0}^{m} x_i \sum_{i=0}^{m} y_i}{(m+1)\sum_{i=0}^{m} x_i^2 - (\sum_{i=0}^{m} x_i)^2} \end{cases} \tag{7}$$

We are only concerned with the k parameter for it is the slope of the fitted line.

3.4. Skew Correction

We use the flowing formula to rotating the bar code image to a level position.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \tag{8}$$

Because the output pixel is often mapped to a non-integer position through the calculation of the image rotation, therefore interpolation must be used to determine the grey value of that positing. Here we use the bilinear interpolation method [3] to fill the output pixels. Bilinear interpolation is often used to improve image quality after performing spatial transformation operations such as digital zooming or rotation. The quality of the corrected image is satisfied because it doesn't lose mush detail.

4. Experiment

We have got 120 samples that have skew angles from -45° to +45° captured by a video camera. The sample images have background and glare interference noise. We applied the Hough transform and our method to detect their skew angles. The detection result is thought to be correct when the detection error is between ±2° because it doesn't influence the decode processing.

Table 1. The experimental results (P4 1.6G, 512M memory)

Method	Success rate (%)	Average speed (ms)
Hough transform	72	760
Our method	80	80

The Hough transform method and our method are compared in terms of their processing time and their accuracy in determining skew angles. As shown in table 1, our method has advantages in both, especially in the processing time, making it acceptable to embedded bar code recognition devices. In the term of detection accuracy, our method did not perform as well as hoped. Although the presence of noise limits the effectiveness of Hough transforms, however, it doesn't influence our method because of the precise image segmentation. In fact, we found that the selecting of the set of eigen-points is the most important factor in the results.

5. Conclusions

The image skew correction is an important step in the processing of 2D bar code recognition. In general, the methods to detect image skew angles are based on Hough transform and projection methods, but these methods are not appropriated for real-time applications because they are computationally expensive. This paper presents a fast method for complex 2D bar code image skew detection. Using the line information of the bar code's borders, this method can search for the borders' skew angle, even when the detected image has high noise. The experiment demonstrated its success. Further research along this direction will focus on the detection accuracy.

Acknowledgements

Parts of the work presented herein were funded by the China Innovation Fund for Small Technology-based Firms (02C26214400224), and GuangDong Provincial Natural Science Funding Project B6-109-497.

References

[1] Stuart C. Hinds, James L. Fisher, and Donald P. D'Amato, "A document skew detection method using run-length encoding and the Hough transform", proceedings of the 10th internationally conference on pattern recognition, Vol. 1, pp. 464-468, 1990.

[2] Q. Zhang, L.Z Ma, Y. Gao, and Z.H. Chen, "Skew detection method for bill image based on directional projection", Computer Applications, Vol. 24, pp. 50-52, 2004.

[3] Kenneth R. Castleman, "Digital Image Processing", Prentice-Hall Inc, pp. 94-114, 2003.

[4] Z.H. Wang, Z. Li, S.J. Cai, "Detection for skew of document image based on least square method", Computer Applications and software, Vol. 13, pp. 43-46, 2001.

[5] Avanindra and Subhasis Chaudhuri, "Robust Detection of Skew in Document Images", Image Processing, Vol. 6, pp. 344-349, 1997.

[6] W. Postl, "Detection of linear oblique structure and skew scan in digitized documents", Proceedings of the 8th International Conference on Pattern Recognition, pp. 687- 689, 1986.

[7] Lawrence O'Gorman, "The document spectrum for page layout analysis", IEEE Transactions on PAMI, 1993, Vol. 15, No. 11, pp. 1162- 1173, 1993.

[8] D.S. Li, G.R. Thoma and H. Wechsler, "Automatic page orientation and skew angle detection for binary document images", Pattern Recognition, Vol. 27, pp. 1325-1344, 1994.

[9] H. Yan, "Skew correction of document images using interline cross-correlation", Computer Vision, Graphics, and Image Processing, Vol. 55, pp. 538-543, 1993.

[10] Srihari, S.N. and Govindaraju, "Analysis of textual images using the Hough transform", Machine Vision, Vol. 2, No. 3, pp. 141-153, 1989.