CONFIDENCE MEASURE FOR SKEW DETECTION IN PHOTOGRAPHED DOCUMENTS


Abstract: Skew detection and correction poses particular challenges in deformed images, as well as images suffering from camera lens distortions or distortions caused by non-flat scanning surfaces. The current paper introduces an algorithm that yields good results on such images and proposes a confidence measurement to estimate the accuracy of the returned skew angle.

Key words: skew, neighbor clustering, confidence measurement

1. INTRODUCTION

In recent years, more and more institutions, companies and libraries have made the transition from plain paper documents and books towards digital content. As many of them retain large paper based archives that need to be integrated in their new digital systems, faster and more robust automated content conversion systems are needed to speed up and cut down on costs for converting books and archives to digital form.

Skew detection and correction for scanned or photographed images is a field of great interest and thus many solutions have been proposed over the years. There are four main classes of algorithms (Baird, H. S. 1995): Hough transform, projection profiling, cross-correlation and neighbor clustering.

The current article presents an approach for detecting the skew angle of an image, with a new method for estimating a confidence level for the detected skew angle. The present skew detection algorithm identifies lines of text. Then, using specific feature-points selected inside a detected line of text, the skew angle is estimated by various line fitting techniques (R. Smith., 1995). (Lu, Y et al., 2003).

2. RELATED WORK

Skew detection and correction for scanned or photographed images is a field of great interest and thus many solutions have been proposed over the years. There are four main classes of algorithms (Baird, H. S. 1995): Hough transform, projection profiling, cross-correlation and neighbor clustering. Hough transform (Chandan Singh et al., 2008) and projection profiling based algorithms are constructed on the presumption that parallel lines of text will generate collinear agglomerations of foreground pixels, but use different mathematical representations of the idea. Cross correlation based algorithms use the general coherency between stripes from lines of text that can be expected in a skew free document.

Neighbor clustering techniques use clustering methods to rebuild lines of text. Then, using specific feature-points selected inside a detected line of text, the skew angle is estimated by various line fitting techniques (R. Smith., 1995). (Lu, Y et al., 2003).

3. ALGORITHM

The present algorithm is based on the observation that good skew features can be extracted from the borders of image entities. For most characters, especially Latin ones, bottom borderlines can provide excellent measurement of skew angles. The present skew detection algorithm identifies lines of text and extracts skew angle information based on these lines. In order to overcome errors introduced by camera lens distortions in images, the current paper introduces a modified skew angle estimation method and a new approximation of the accuracy of results.

3.1 Preprocessing

If the image is not bitonal, the image will first be binarized, using an existing algorithm, like (J. Sauvola, 2000). Entities are then discovered and contour information is preserved.

3.2 Noise removal and entities filter

The main filter in the present work is a minimal size threshold used to remove noise. Considering the 6 point letter ‘e’ as a minimum size reference, it can be shown that most Latin characters are not readable below this size. Thus, entities below 6 pixels can be viewed as noise.

Considering that the human eye cannot distinguish, unaided, details smaller than 0.1 mm, entities below 0.6 mm of height can be removed from the computation. Thus, the minimum accepted height, measured in pixels, will be (considering that an inch has 25.4 mm):
\[
\text{noise\_threshold} = \text{round}\left(\max\left(0.6n \times \text{Image\_ROI}^2, 25.4\right)\right).
\]

A second heuristic threshold is used for large entities removal. If too much of the input entities are removed, the second threshold can be reduced in order to retain more input data. Further filtering is done within the clustering algorithm.

3.3 Cluster construction

Clusters are initially constructed based on a vicinity clustering algorithm that is optimized for horizontal Latin based text.

A character's vicinity is defined as rectangular area: of 120% height, and of a length of 200% of the character's height, starting from its right bottom pixel. A maximum height difference of 200% is also used. This character clustering algorithm yields good results in the presence of skew angle in the range ±15° and normal camera lens distortions.

3.4 Cluster selection

It is important that the clusters offering better skew information are retained, while the clusters that can introduce errors are discarded.

Clusters of small length will generally not retain enough input information for an accurate skew detection. However, depending on the page layout, there are pictures that do not contain sufficiently long lines. Thus, cluster filtering will be tried in an iterative fashion, from highest precision, until a sufficient number of clusters are found:

\[\text{cluster\_threshold} = \frac{1}{\tan(i \times \text{angular\_precision})}\]

where \(i\) is the iteration number and \(\text{angular\_precision}\) is the desired skew angle detection accuracy.

3.5 Skew angle estimation

For each cluster, two approximate skew angle computations are made by means of a least square fitting method performed on the bottom pixels of each entity in the cluster. The first is used to find the direction of the skew fault, and the second is used to find the angle, while removing descenders in the computation.

Next, the input points are rotated towards the OX axes with the determined skew angle.

Finally, the data points are fitted to a parabolic function in order to determine the straightness of the cluster. There are two indicators of the quality of a cluster for skew detection: cluster length and cluster "flatness" – determined by the distance between the focal point and the extreme of the parabolic function.

Thus, considering that the cluster length in pixels is \(\text{cluster\_length}\) and that the fitting function was determined as:

\[a x^2 + b x + c\]

a confidence degree can be defined as:

\[\text{confidence\_degree} = \text{cluster\_length} \times a.\]

3.6 Final skew angle computation

A histogram is used to eliminate values that are too far off the main peaks. The final skew angle computation takes into account the confidence of each cluster, and is thus computed as a weighted mean between the skew angle and the confidence degree for each cluster.

\[\text{final\_angle} = \frac{\sum \text{angle}[i] \times \text{confidence\_degree}[i]}{\sum \text{confidence\_degree}[i]}\]

3.7 Cluster recomputation

After the final skew angle was computed, if the input data quality is considered of low quality, a new computation can be made, by recomputing clusters using a modified version of the vicinity clustering that account for the direction and the approximate angle of the skew fault.

Finally, steps 3.4 to 3.7 are reiterated and the final skew angle is returned.

4. DEGREE OF CONFIDENCE OF FINAL RESULTS ESTIMATION

As the next steps in an automated digitalization process depend on the results obtained by the skew detection algorithm, a degree of confidence is introduced as a second returned value for the presented algorithm.

The expected error is inverse proportional to the length of the clusters and with the straightness of the detected parabolas.

Thus, the algorithm will return two values:
- the skew angle, in degrees
- a confidence measure

The confidence measure is given by the formula:

\[\sum \text{confidence\_degree}[i] / \text{number\_of\_clusters}\]

In the present form, the confidence measure is not normalized, but represents a way of evaluating the quality of the returned value by comparing the confidence measure returned for various images.

Moreover, in systems that use feedback loops, an image can be processed several times, each time with different pre-processing filters, and the confidence measure can be used to determine which filters yield the best results.

5. FURTHER WORK

For now, the algorithm achieves great accuracy and provides a measure of the degree of accuracy for the returned skew angle, depending on the quality of the input data.

Further work will be done in order to obtain a normalized degree of confidence that will likewise be returned together with the skew angle, but providing a better measure of the skew detection success. Two values shall be returned.

First, the algorithm will return the angle accuracy, measured in degrees, which will represent the maximum possible error in the returned angle, and thus an angle space in which the true skew angle will surely reside.

The second value will be a percentage indicating the probability that the returned skew angle is the actual true skew angle.

6. ACKNOWLEDGMENTS

The research presented in this paper is supported by the national project “Excelență în cercetare prin programe post doctorale în domeniul prioritar al societății bazate pe cunoaștere (EXCEL)”, Project POSDRU/89/1.5/S/62557.

7. REFERENCES


