Fast Seamless Skew and Orientation Detection in Document Images

Iuliu Konya Stefan Eickeler Christoph Seibert
Fraunhofer Institute for Intelligent Analysis and Information Systems (IAIS)
Schloss Birlinghoven, 53754 Sankt Augustin, Germany
{Iuliu.Konya, Stefan.Eickeler, Christoph.Seibert} @iais.fraunhofer.de

Abstract

Reliable and generic methods for skew detection are a necessity for any large-scale digitization projects. As one of the first processing steps, skew detection and correction has a heavy influence on all further document analysis modules, such as geometric and logical layout analysis. This paper introduces a generic, scale-independent algorithm capable of accurately detecting the global skew angle of document images within the range [-90°, 90°]. By using the same framework, the algorithm is then extended for Roman script documents so as to cope with the full range [-180°, 180°) of possible skew angles. Despite its generality, the improved algorithm is very fast and requires no explicit parameters. Experiments on a combined test set comprising around 110 000 real-life images show the accuracy and robustness of the proposed method.

1. Introduction

Document skew is a very common distortion found in document images as a result of the digitization process, or as a feature of the document's layout. In most cases, it has a detrimental effect on the accuracy of the subsequent geometric and logical layout analysis steps. This is due to the fact that most existing algorithms require a proper document alignment before application, although there exist a few methods which do not require any previous skew correction (e.g. [13]). However, the skew-independent layout analysis methods either pose certain restrictions on the possible angle range, or are considered in isolation from the subsequent processing operations, such as region classification or text line extraction. As a result, this apparent simplification leads to the necessity of applying more sophisticated techniques for other tasks.

In general, the skew within a document page can fall into one of the following three categories: global skew, assuming that (almost) all page blocks have the same slant; multiple skew, when certain blocks have a different slant than the others; non-uniform text line skew, when the slant fluctuates (such as lines having a wavy shape). In the current paper, we will only be concerned with global skew estimation as the most prevalent type of skew present in document images. The interested reader may find a recent algorithm dealing with non-uniform text line skew in [16], and a more detailed survey of algorithms capable of dealing with both global and multiple skew in [4, 5, 8, 11]. The largest classes of methods for skew detection are based on projection profile analysis, Hough transform or nearest-neighbor clustering. Recently the focus of the research community has shifted more towards the nearest-neighbor approaches, as they seem to offer the greatest flexibility and accuracy.

For the processing of large document collections however, the limited skew angle detection offered by classical algorithms is insufficient, as documents may have any possible orientation. One of the earliest algorithms able to distinguish between portrait and landscape orientations was proposed by Akiyama et al. [2] in 1990 for Japanese documents. About at the same time, algorithms using the ascender-to-descender ratio for Roman script documents were introduced for detecting the up/down orientation [17]. In the recent years a few algorithms capable of detecting both skew and orientation for Roman script document images have appeared [4, 18]. For documents making heavy use of majuscules or using scripts other than Roman (e.g. Pashto, Hindi, Arabic), different decision criteria for orientation detection were recently introduced in [3].

In the current paper we describe two new algorithms for global skew detection using the same generic framework. The algorithms belong to the class of nearest-neighbor approaches, as they rely on the construction of a Euclidean minimum spanning tree (MST). The basic, script-independent algorithm is able to deal with document images having a global skew angle within

1051-4651/10 $26.00 © 2010 IEEE
DOI 10.1109/ICPR.2010.474

IEEE computer society

$[-90°, 90°]$. An improved version of the algorithm specializes on Roman script documents and can work with document skew from the full angle range of -180 to 180 degrees, while at the same time being more accurate. Both algorithms are very fast, their runtime speed being directly comparable to that of the early projection profile-based methods [6]. An important feature contributing to the robustness and speed of both algorithms is the fact that, in contrast to other recent approaches [4, 18], they do not require any kind of layout analysis, such as text line determination. As a direct consequence of this fact, no explicit parameters are needed in either algorithm. Extensive testing on a total set of about 110 000 images was used to validate the proposed technique and compare it to other state-of-the-art methods.

2. Methodology

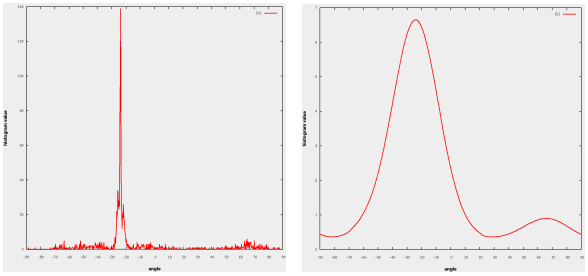
We assume as input to our algorithm a binary document image. A wide selection and performance comparison of binarization algorithms may be found in [10], describing the results of an international document image binarization contest. For simplicity reasons we will also consider that a list containing all connected components forming the foreground of the given document image is also available. The labeling of the connected components from a binary image can be accomplished efficiently using any standard algorithm (e.g. [9]). Since most existing techniques for geometric and logical layout analysis of documents also require a binarized image as well as the list of connected components [8], there is practically no/little extra effort necessary to produce the required input data.

2.1. Skew Detection

In order to be able to robustly deal with generic document images containing non-text regions of significant size, the first step is a basic filtering of potential characters/character parts from non-text content. Note that for document images known not to contain halftones or graphics this step can be skipped entirely. The foremost purpose of the filtering is to reduce the amount of noise in the image, be it impulse-like or small connected components from dithered halftones/drawings. Two simple filtering rules were applied for each connected component throughout all our experiments:

1. The width and height do not differ by more than a factor of 10. The relatively large threshold ensures that thin letters (such as "i" or "l") are kept, along with any components consisting of a few wrongly merged characters (possibly as an artifact of the binarization process).

2. The width and height are larger than certain thresholds. The thresholds in our case were set to 50% of the size of the dominant character width/height on the page, determined as described in [14]. In this way the thresholds are resolution-independent. Note that since neither the orientation nor the skew of the page are known at this stage, the width and height thresholds must be used both normally and interchanged in the condition.



a) b)

Figure 1. Binned histogram of MST skew angles: a) original; b) convolution result

Next, we take the centers of the bounding boxes of the filtered connected components and compute the Euclidean MST of this set of points. It was experimentally determined that for an unrestricted skew range, the center points provide a more robust estimate of the global skew than both the upper and lower mid-points of the bounding boxes. The Euclidean MST for a set of points can be computed either directly [1] or indirectly by using the property that the MST edges are a subset of the edges of the Delaunay triangulation and determining the respective subset by any standard MST algorithm, such as that of Kruskal. A worst-case running time of $O(n \log n)$, where n represents the number of points, is achievable by using either variant.

A binned histogram spanning the angle range $[-90°, 90°)$ is computed from the skew angles of the MST edges. In our experiments we have used a bin size of $0.1°$, as skew angles lower than this value are practically indistinguishable by humans.

The final step is the detection of the skew angle from the computed histogram. Unlike other binned histogram-based methods [6] we do not employ an iterative coarse-to-fine search method, as we have found it to be too sensitive at larger skew angles (i.e. $\geq$15–20°). Instead, we assume that the skew angle errors are normally distributed around the global skew angle and convolve the histogram circularly with a Gaussian mask. Thus, the location of the maximum value in the

result of the circular convolution is expected to correspond to the desired skew angle. In our experiments we have determined that a Gaussian mask diameter equal to the number of bins corresponding to 90° in the histogram performs well.

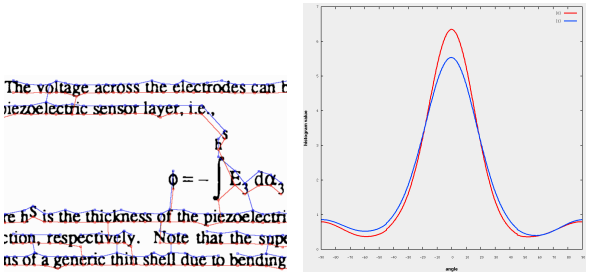
For obtaining a higher/lower accuracy, the histogram bin size can be easily reduced/extended, having a direct influence on the running time of the algorithm. This is especially visible for very fine-grained histograms, where the $O(n^2)$ convolution time will be the dominant factor in the overall running time of the algorithm. At this point it is important to note that the algorithm presented in this section is script-independent, since its sole indirect assumption is that character spacing is usually smaller than line spacing.

2.2. Improved Skew and Orientation Detection

To the best of the authors' knowledge, the idea of using an Euclidean MST for determining the orientation of text lines was first introduced by Ittner and Baird in [12]. However, because of the fact that their method only made use of the center points for each connected component, it was inherently unable to differentiate between top-left and top-right orientations.

In contrast to Ittner and Baird's method and the previously presented algorithm, the method proposed in the following is specific to Roman script documents. More specifically, it makes indirect use of the fact that in a typical text, the number of descenders is lower than the number of ascenders. This holds true for many languages having Roman script, such as English, German, Spanish and French. One may easily verify this fact from existing tables containing character frequencies for each language, such as the ones for the English language in [3].

As a first step of the improved algorithm we use the result provided by the basic algorithm as an approximation of the actual page skew. Next, for each of the filtered connected components we compute the top-left and bottom-right coordinates of its bounding box in the document image deskewed using the approximated skew. This can be accomplished by applying the appropriate, pre-computed rotation matrix to each pixel within a connected component. Note that it is not actually necessary to consider each pixel of a component, just the pixels located on its border. The processing time gain by applying this trick has been found to be insignificant, however. We now compute two sets of points, namely the top and bottom mid-points for each component. Afterwards, the same processing as in the basic algorithm is applied on each of the two sets of points and two histograms convolved with Gaussian masks are



a) b)

Figure 2. Portion of document image with superimposed Euclidean MSTs and their corresponding histograms

obtained. One may now make use of the ascender-to-descender ratio property to conclude that the histogram containing the higher maximum value corresponds to the actual alignment points (bottom mid-points) of the characters on the page. This is true because the histogram of the alignment point edges has more angles close to the page skew and their distribution will naturally feature a taller and sharper peak. Thus the correct up/down orientation of the page is determined. Finally, we can improve the initial approximation of the page skew by selecting instead the bin containing the maximum value from the histogram of approximated alignment points as the one corresponding to the desired result.

3. Experimental Results

For testing the orientation detection accuracy of our algorithm, we have used 5 different test sets. In all our experiments we have differentiated between 4 different orientations: top-up, top-down (180°), top-left (90° counter-clockwise) and top-right (90° clockwise).

The first test set consisted of the 979 test images from the UW-I dataset [15], containing technical journal scans featuring 1, 2 or 3 layout columns. Our second test set was the OCRopus test set [18], consisting of 9 different images, each scanned in four different resolutions, namely 150, 200, 300 and 400 dpi. Each image was also rotated in all four orientations, thus resulting in a total number of 144 test images. The third and fourth test sets consisted of 100 single-column, respectively 109 two-column journal images, obtained by converting the PDF version of several recent articles into 300 dpi raster images. Each of the images in these test sets was rotated 360 times using bicubic interpolation with an interval of 1°, starting from -180°, thus totaling 36 000, respectively 39 240 images. The last test set consisted of a uniform sample of 100 images from the UW-I test set, selected manually from their "skew-

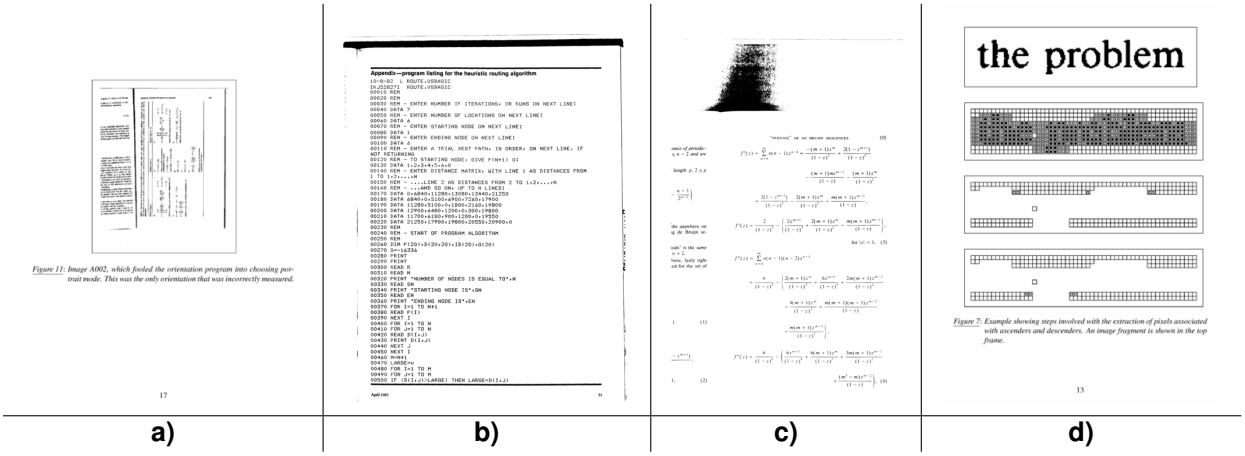


Figure 3. a) Document image with the maximum error in skew angle detection due to a contained document image reproduction; b-d) Orientation detection failures due to: all-uppercase listings, math formulas or graphic objects arranged such that the character spacing is higher than the line spacing

free" version, available as part of the UW-III database. Again, each of the images was considered in 360 different rotated variants, leading to a test set size of 36 000 images. Since the "skew-free" variants of the UW-I images were obtained by the UW creators using an automatic skew detection and correction algorithm, their real skew had to be checked manually and those images were selected where the difference to the real skew was at most about 0.1°. Note that since the scans feature wavy lines, distortions near the bookfold and significant skew differences among the different layout columns of the same document, the slant of the text lines even within the same page frame varies considerably – up to about 0.42° in our selection.

In order to evaluate the skew detection accuracy we have used the last three datasets, as they are the only ones large enough to obtain meaningful results. For computing the skew detection error for each image we have considered that the orientation was detected correctly, i.e. the skew angle for each orientation falls between [-45°, 45°). This has allowed us a direct accuracy comparison with the proposed basic algorithm on the complete data sets.

On the UW-I test set our algorithm performed very well and failed only on images containing almost exclusively majuscules, digits and/or mathematical formulae. As for all algorithms relying on the ascender-to-descender ratio, such documents are inherently impossible to classify correctly. In such case, a combination with methods using other orientation-dependent features (e.g. [3]) is necessary.

For the OCRopus test set, the proposed orientation detection achieved a 100% success rate, the same as

Table 1. Orientation detection results on the UW-I dataset

UW-I	Total images	Bloomberg et al. [7] # correct	van Beusekom et al. [18] # correct	Proposed # correct
top-up	970	935	963	966
top-left	9	2	7	7
top-down	0	0	0	0
top-right	0	0	0	0
Total		95.8%	99.1%	99.4%

the algorithm proposed by van Beusekom et al. [18] and better than the Bloomberg et al. method [7] with 93.8%. The perfect result was possible because the test set consists exclusively of text-only, artificial images. Note that the main purpose of this data set was to test the resolution independence property without the influence of other factors.

Table 2. Skew and orientation detection results on datasets 3–5

	Total images	SSkewDet Basic			SSkewDet Improved			
		avg. err.	std. dev.	max. err.	avg. err.	std. dev.	max. err.	orient. errors
1 col. PDF	36 000	0.21	0.28	1.95	0.08	0.13	1.2	84
2 col. PDF	39 240	0.22	0.28	1.65	0.09	0.14	1.0	0
UW-I selection	36 000	0.26	0.32	1.55	0.15	0.2	1.0	0

From table 2 one may see as expected that the accuracy of the improved algorithm was indeed better that

that of the basic algorithm. The rising average erorrs directly reflect the increasing difficulty of the data sets. In case of the UW-I subset the sharp rise of the average errors and their corresponding standard deviations also points at the inaccuracy of the "ground truth". Overall, the accuracy of both algorithms was very good, being similar to that of algorithms having constraints on the logical layout [4]. In fact, the only observed errors produced by the orientation detection algorithm were for documents where its preconditions were violated (as seen in figure 3).

The processing times on a 300dpi A4 document image (approx. 2500x3500 pixels) are 0.01 seconds for the basic algorithm and 0.11 seconds for the improved version. For a 200 dpi image the times are 0.01 seconds and 0.07 seconds, respectively. The computer used for the tests had a Core2Duo 2.66Ghz processor. In comparison, the fastest state-of-the-art algorithm reported around 0.01 sec processing time on 200dpi images [4] on a relatively similar computer configuration.

4. Conclusion

In this paper we proposed two new algorithms for global skew detection, both relying on the construction of the Euclidean MST. The robustness and accuracy of the algorithms was shown on several large, real-life data sets. The results obtained on the public data sets are better than those of other state-of-the-art methods. Both algorithms have very good run-time performance and are simple to implement as they do not require any kind of logical layout segmentation.

A further evaluation on a grayscale/color document image dataset with respect to different binarization/color reduction algorithms would be of great interest, as this would be more in line with the current demands for real-life document digitization projects. Unfortunately to the authors' best knowledge no large datasets suitable for this task currently exist.

References

[1] P. Agarwal, H. Edelsbrunner, O. Schwarzkopf, and E. Welzl. Euclidean minimum spanning trees and bichromatic closest pairs. *Discrete and Computational Geometry*, 6(6):407–422, December 1991.

[2] T. Akiyama and N. Hagita. Automated entry system for printed documents. *Pattern Recognition*, 23:1141–1154, 1990.

[3] H. Aradhye. A generic method for determining up/down orientation of text in roman and non-roman scripts. *Pattern Recognition*, 38(11):2114–2131, 2005.

[4] B. T. Ávila and R. D. Lins. A fast orientation and skew detection algorithm for monochromatic document images. In *DocEng '05: Proc. ACM symposium on Document engineering*, pages 118–126, New York, NY, USA, 2005. ACM.

[5] A. Bagdanov and J. Kanai. Evaluation of document image skew estimation techniques. *Document Recognition III*, 2660:343–353, 1996.

[6] H. Baird. The skew angle of printed documents. In *Proc. Conf. Society of Photographic Scientists and Engineers*, volume 40, pages 21–24, Rochester, NY, 1987.

[7] D. Bloomberg, G. Kopec, and L. Dasari. Measuring document image skew and orientation. In *Pro. SPIE Document Recognition II*, pages 302–316, San Jose, USA, 1995.

[8] R. Cattoni, T. Coianiz, S. Messelodi, and C. Modena. Geometric layout analysis techniques for document image understanding: a review. Technical Report 9703-09, ITC-irst, 1998.

[9] M. B. Dillencourt, H. Samet, and M. Tamminen. A general approach to connected-component labeling for arbitrary image representations. *J. ACM*, 39(2):253–280, 1992.

[10] B. Gatos, K. Ntirogiannis, and I. Pratikakis. ICDAR 2009 document image binarization contest (DIBCO 2009). In *Proc. Int'l Conf. Document Analysis and Recognition (ICDAR)*, pages 1375–1382, 2009.

[11] J. Hull. Document image skew detection: survey and annotated bibliography. *Document Analysis Systems*, 2:40–64, 1998.

[12] D. J. Ittner and H. S. Baird. Language-free layout analysis. In *Proc. Int'l Conf. Document Analysis and Recognition (ICDAR)*, pages 336–340, 1993.

[13] K. Kise, A. Sato, and M. Iwata. Segmentation of page images using the area Voronoi diagram. *Computer Vision and Image Understanding*, 70(3):370–382, 1998.

[14] I. Konya, C. Seibert, S. Eickeler, and S. Glahn. Constant-time locally optimal adaptive binarization. In *Proc. 10th Int'l Conf. Document Analysis and Recognition*, pages 738–742. IEEE, 7 2009.

[15] I. Phillips. User's reference manual for the UW english/technical document image database III. Technical report, Seattle University, Washington, 1996.

[16] A. Spitz. Correcting for variable skew in document images. *Int. Journal on Document Analysis and Recognition (IJDAR)*, 6:192–200, 2004.

[17] S. Srihari and V. Govindaraju. Analysis of textual images using the Hough transform. *Machine Vision and Applications*, 2(3):141–153, June 1989.

[18] J. van Beusekom, F. Shafait, and T. M. Breuel. Resolution independent skew and orientation detection for document images. In *DRR*, pages 1–10, 2009.