

Rotation Angle Estimation of Scanned Handwritten Cursive Text Documents

Ivo R. Draganov¹, Antoaneta A. Popova²

Abstract – In this paper we propose an algorithm that estimates the rotation angle presenting in scanned images of handwritten cursive text. It combines a technique of tracking the separate words connectivity, narrowing the range of possible rotation angles to be checked and the extra obtained results are found useful in the following stages of a complete handwritten text recognition system.

Keywords – handwritten text recognition, angle estimation

I. INTRODUCTION

It is a common task to check if the input image in most of the optical character recognition (OCR) systems for both printed and handwritten text has non-zero rotation angle. If this is so it is indication for either the sheet of paper has not been correctly aligned in the scanner's bed and or it is an effect due to the native characteristics of the writer handwriting or different styles for the text lines used. The rotation angle for the whole image (page) in all these cases should be correctly estimated and a new image with the right alignment used in the further processing steps.

There are numerous approaches trying to solve the problem mentioned above. We narrow it concerning only images containing handwritten cursive text. In general – the angle between every text line and the base of the image could be different, as well as the angle between any pair of the lines. Furthermore any line may contain certain linear parts that form different angles with the image baseline or speaking in other words – there may be curved lines. This makes the problem even harder to solve.

One of the very first attempts to solve this problem concerning printed text documents was done by Glauberman [1]. In the early days of the OCR systems he suggested the use of horizontal projections line-by-line counting the number of pixels forming the text in the image and thus forming the vertical histogram. Getting all the histograms for different angle projections (in a certain range with a given step) and finding the one corresponding to the minimal entropy directly gives us the rotation angle we are looking for. This basic principle lies down in great number of other approaches and usually leads to very good results.

Pal et al. [2] propose a technique for multi-oriented text lines detection and their skew estimations. They use grouping boxes around single characters which help them to determine their belonging to certain words. Afterwards they find key points from the detected words which form reference lines. These lines then are used to extract whole lines of text recognizing the accurate angle to which each line is rotated. Artistic effects are not considered as a serious barrier in front of this approach.

In [3] Shivakumara et al. use static and dynamic thresholds in the process of separation every text line from the image including the usage of the projection profile. They also use linear regression analysis to estimate the skew angle for each text line after the separation. This approach is considered to be a step ahead including the Glauberman principle and optimizing the whole process of segmentation the lines from the image.

Clark and Mirmehdi in [4] also use projection profiles but here they are introduced to locate the horizontal vanishing point of the text plane. Thus they segment the lines of text and then reveal the style of justification of the paragraphs. Vertical vanishing point is obtained from analyzing the change in line spacing. This method is another serious step in estimating rotation (of the whole document) and skew (of the separate lines) angles and originally applicable for images of documents taken by a camera (not only a scanner) at different angles and without the knowledge of the focal length.

A final example for the expansion of the projection profiles in finding the rotation and skew angles and the lines separation as well, is given in [5]. Messelodi and Modena propose there the usage of a great number of heuristic filtering rules, as well as the coordinates of the centers and vertexes of the bounding rectangles around every candidate line, word and character. They implement three-level alignment segmentation for every line candidate which leads to good results even in images taken from complex real scenes containing some sort of text (printed, handwritten, with artistic effects etc.).

Our goal is to extend the basic algorithm using projection profiles [1] for finding the rotation angle of the whole page in such way that it can perform faster and in the same time can process images containing handwritten text which lines are not straight. The last property is absent in the most algorithms proposed so far for solving this problem.

In the next second section of this paper we describe such an algorithm and in the third one we give some experimental results. Afterwards we make our conclusions in the fourth part.

¹Ivo R. Draganov is with the Faculty of Communications and Communications Technologies, Technical University, Kliment Ohridski 8, 1000 Sofia, Bulgaria, E-mail: idraganov@abv.bg

²Antoaneta A. Popova is with the Faculty of Communications and Communications Technologies, Technical University, Kliment Ohridski 8, 1000 Sofia, Bulgaria, E-mail: antoaneta.p@komero.net

II. PROPOSED ALGORITHM

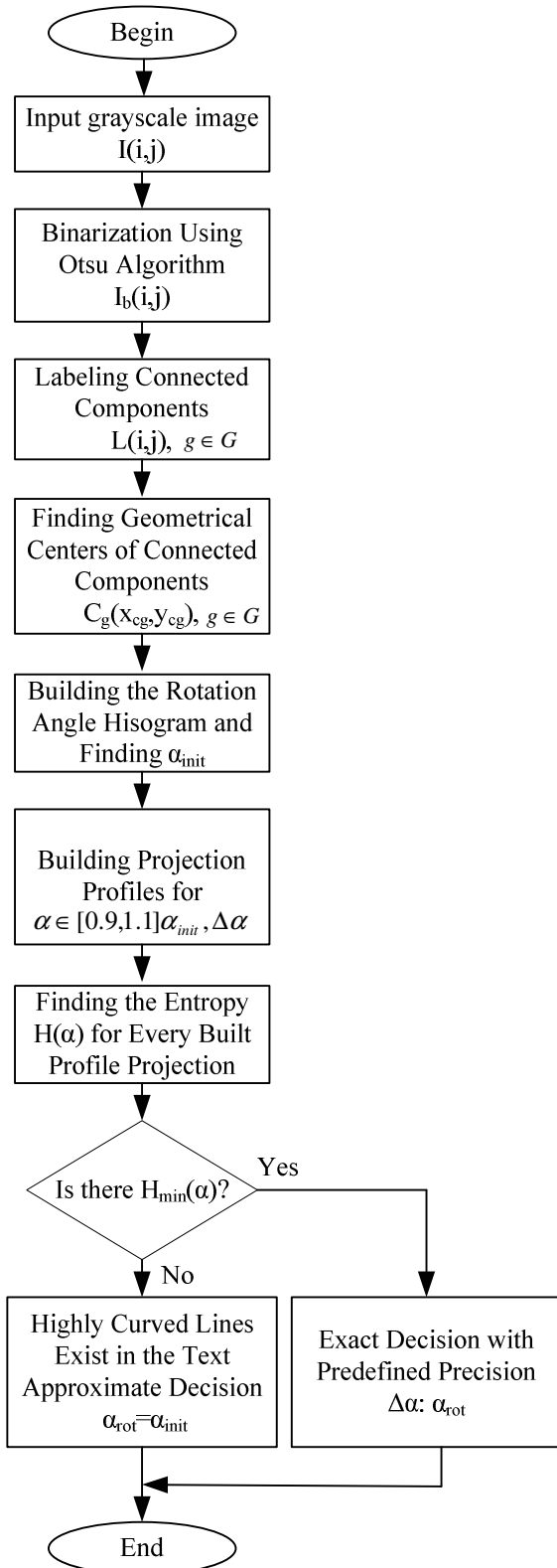


Fig. 1. Flowchart of the proposed algorithm

Narrowing the possible input only to handwritten text and knowing the advantages of the Messelodi-Modena method,

described in [5], we propose a faster algorithm (Fig.1) due to the limited input mentioned. One heuristic rule is used concerning the values of one threshold, as well as we do not need to build a boundary rectangles forming line, word and character blocks and perform three-level segmentation. As for the word/character segmentation we suggest using of an indexing (labeling) algorithm which has proven its efficiency. Thus suggested algorithm can be described in the following steps:

1) Input a grayscale image, described by the intensity function $I(i, j) \in [0, 255], i \in [0, M-1], j \in [0, N-1]$ containing the handwritten text scanned. We assume that the sheet of paper containing the original text has no background elements (e.g. stripes) and no strikes over the text as well.

2) Applying intensity segmentation with one threshold (binarization) using Otsu algorithm:

$$I_b(i, j) = \begin{cases} 1, & I(i, j) \geq \theta_{opt} \\ 0, & I(i, j) < \theta_{opt} \end{cases}, \quad (1)$$

for $i \in [0, M-1], j \in [0, N-1]$. θ_{opt} - the optimal intensity threshold estimated by the Otsu algorithm.

3) Labeling (indexing) all elements (pixels) in the image (i, j) with a number that differs for elements belonging to different words or separate characters. We use the mask shown on Fig. 2a).

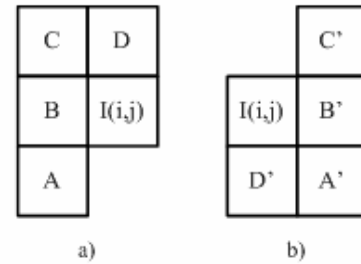


Fig. 2. Working masks used for labeling

Using the four neighbour elements A, B, C and D we label the current element (i, j) if its value is 1 using:

$$L(i, j) = \min\{L_1(A), L_2(B), L_3(C), L_4(D)\}, \quad (2)$$

where $L(i, j)$ is the index associated with (i, j) . Then we make a table of connectivity as shown in Table I. It shows the label of every element and with which other elements is connected. The first scan should be made vertically from top to bottom and from left to right.

 TABLE I
TABLE OF CONNECTIVITY

g-th Label	Connected with m(g) marks
2	3
3	2
4	5, 6
...	...

G is the maximal number of objects in the image. $m(g)$ is the label for the g -th object from all the G objects.

A second scan should be made using the mask from Fig. 2b) and it has to be done in bottom-top and right-left directions again vertically. This second labeling is actually a checking function and in case that after is done some objects from the left column g -th Label in Table I have greater value than the labels of their connected components from the right column $m(g)$ it is necessary these labels to be exchanged. Thus it is possible some numbers already associated with some objects to become redundant.

4) At this step the geometrical centers are found for every labeled object, using:

$$x_{cg} = \frac{\sum_{p=0}^{P-1} x_{pg}}{P}, y_{cg} = \frac{\sum_{p=0}^{P-1} y_{pg}}{P}, \quad (3)$$

where g is the label of the respective object, P is the number of pixels that it consists of and c stands for center. x_{cg} and y_{cg} are the coordinates of the geometrical center $C_g(x_{cg}, y_{cg})$ for the g -th object.

5) At this step we use a distance threshold l_{opt} . It represents the average Euclidean distance $d_E(C_g, C_{g+1})$ between the centers of connected components which are neighbours from a single text line and its value is estimated experimentally over a large number of test images. Having this threshold we estimate the angle α formed by the line connecting each two centers from step 4), the distance between which is smaller than l_{opt} , and the base of the image. The process is illustrated on Fig.3. Eq. (4) represents it more clearly:

$$\alpha = \begin{cases} \arctg\left(\frac{y_{c(g+1)} - y_{cg}}{x_{c(g+1)} - x_{cg}}\right), & d_E(C_g, C_{g+1}) < l_{opt} \\ \emptyset, & \text{otherwise} \end{cases} \quad (4)$$

All estimated values for α are saved for the next step.

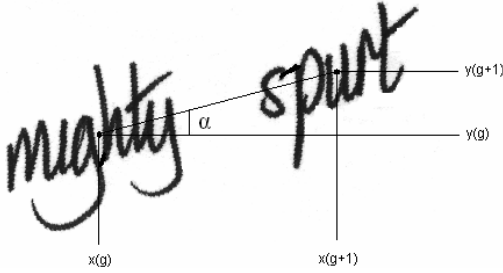


Fig. 3. Estimation of a local value for α defined by connected components centers

6) With all the values for α we can make the histogram of its distribution (Fig.4.b)). Then the global maximum is found and estimated the initial value for the angle to start testing hypothesizes for the accurate rotation angle of the whole image:

$$\alpha_{init} = f \cdot \alpha_{max}, \quad (5)$$

where f is a factor, typically $f=0.9$.

7) At this point we use $\Delta\alpha$ - the step that defines the error in rotation angle estimation α_{rot} . Typically $\Delta\alpha = 0.5^\circ$ but it can be 0.1° as well in some cases where high accuracy is needed. Given this angle step we find the vertical projection profiles

(Fig. 4c)) around the local maximum for α ($\alpha \in [0.9, 1.1] \alpha_{init}$) and the profile with a minimal entropy is the one

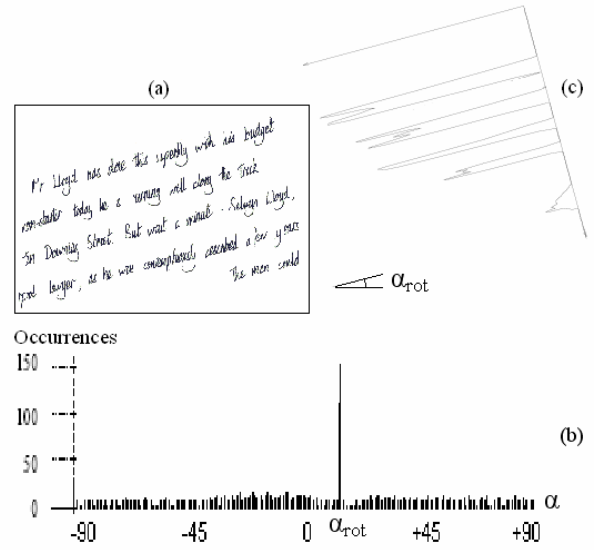


Fig. 4. Rotated text (a), estimation of the α -distribution (b), finding the histogram with minimal entropy corresponding to the angle of initial rotation (c) corresponding to α_{rot} which is the result we are looking for so far. This is done using Eqs. (6) and (7):

$$h_n^\alpha(k) = \frac{h^\alpha(k)}{\sum_{k=0}^{M-1} h^\alpha(k)}, \quad (6)$$

which is the projection profile presented by a normalized histogram (the local value for a single line) for an angle α , k is the current line in the image of total M number of lines;

$$H(\alpha) = \sum_{k=0}^{M-1} -h_n^\alpha(k) \log_2 h_n^\alpha(k), \quad (7)$$

and $H(\alpha)$ is the entropy estimated from that projection profile. The minimum is reached when $\alpha = \alpha_{rot}$.

Finishing the last seventh step the goal is accomplished. It is worth noting that:

- With the estimation of α_{init} we strongly reduce the necessary number of projection profiles. Given $\Delta\alpha = 0.5^\circ$ their number may be less than 5. Otherwise if we check all the angles from -90° to $+90^\circ$ with a step of 0.5° we need to estimate 361 projection profiles!
- If there are highly curved lines or such with alternatively changing slopes in different directions in the text it is obvious that no minimum for $H(\alpha)$ will be reached as for the case of the basic algorithm described in [1]. But α_{init} is a good approximate value for initial rotation angle correction of the whole page before any additional analysis being made for the exact structure of the document if needed.
- Segmenting the connected components and labeling them at this early stage is in great use in the later stages where determining the whole line positions, features extraction and the real recognition process take place.

Of course this labeling comes at a price of using some computing time but compared to the time period needed for estimating 361 projection profiles it is completely excusable.

III. EXPERIMENTAL RESULTS

For our experimentation we used a single page (A4 format white paper) containing handwritten with a black pen text of a single individual with 17 lines present. Then the sheet was scanned for 25 different positions of the rotating angle – from -6 to +6 degrees with a step of 0.5 degrees at 300 dpi resolution with a typical office flatbed scanner in a grayscale (256 gray levels used) mode. The initial estimating of α for non-rotated sheet (0°) showed $+1.5^\circ$ (due to the slope for the separate text lines originally introduced by the writer), and all the other 24 results showed correct estimation of α containing this initial line slope (the results were from -4.5° to $+7.5^\circ$ with a step of 0.5°), so 100% accuracy was reached, but it is obvious that it would be lower if some text with highly curved lines is processed.

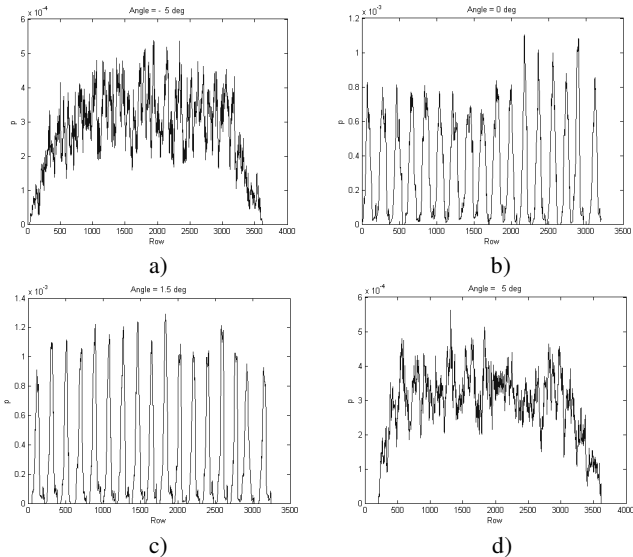


Fig. 5. Different profile projections: a) $\alpha=-5^\circ$, b) $\alpha=0^\circ$, c) $\alpha=\alpha_{rot}+1.5^\circ$, d) $\alpha=+5^\circ$

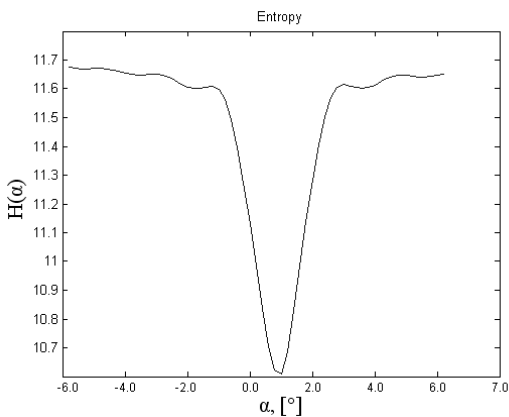
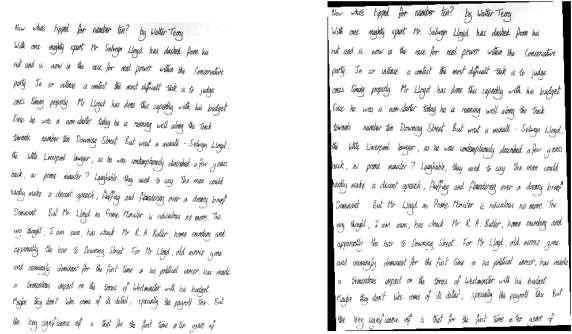


Fig. 6. The entropy as a function of α , reaching its minimum for $\alpha_{rot} = +1.5^\circ$

In Fig. 5 are presented 4 different profile projections. Fig. 5 c) corresponds to the minimum of the entropy.

The entropy $H(\alpha)$ itself is presented in Fig.6 as a function of the angle α .

And finally, Fig. 7a) represents the original input image of the analyzed handwritten text and Fig. 7b) represents an image with the same text but with corrected rotation angle. We used nearest neighbour approximation for maximal fast performance at the stage of rotating the image.



a) b)
Fig. 7. The original input image (a) and the corrected one after rotation the original to $\alpha=\alpha_{rot} +1.5^\circ$ (b)

IV. CONCLUSION

The proposed algorithm in seven steps produced very good results estimating extremely precisely the rotation angle of the whole text document image. It radically reduces the computational time for completing this task in comparison to direct projection profiles method. Moreover it extracts important additional information about the isolated words and characters in the text which will be used in the later stages of a complete text recognition algorithm. Thus the realization of this stage of preprocessing the input image can also be described as a module which can be easily embedded into large and complex systems aiming handwritten text recognition.

REFERENCES

- [1] M. Glauberger, Character Recognition for Business Machines, *Electronics*, No. 29, pp. 132–136, 1956.
- [2] U. Pal, S. Sinha, B. Chaudhuri, “Multi-Oriented Text Lines Detection and Their Skew Estimation”, *Third Indian Conference on Computer Vision, Graphics and Image Processing, Conference Proceedings*, Ahmedabad, India, 2002.
- [3] P. Shivakumara, G. Kumar, D. Guru, P. Nagabhushan, “Skew Estimation of Binary Document Images Using Static and Dynamic Thresholds Useful for Document Image Mosaicing”, *National Workshop on IT Services and Applications, Conference Proceedings*, New Delhi, India, 2003.
- [4] P. Clark, M. Mirmehdi, “On the Recovery of Oriented Documents from Single Images”, *Advanced Concepts for Intelligent Vision Systems, Conference proceedings*, pp. 190-197, Ghent, Belgium, 2002.
- [5] S. Messelodi, C. Modena, “Automatic Identification and Skew Estimation of Text Lines in Real Scene Images”, *Pattern Recognition*, Vol. 32, No. 5, pp. 789-808, 1999.