An Approach to Modification of Water Flow Algorithm for Segmentation and Text Parameters Extraction

Darko Brodić[1] and Zoran Milivojević[2]

[1] University of Belgrade, Technical Faculty Bor, V.J. 12, 19210 Bor, Serbia
dbrodic@tf.bor.ac.rs
[2] Technical College Niš, Aleksandra Medvedeva 20, 18000 Niš, Serbia
zoran.milivojevic@vtsnis.edu.rs

Abstract. This paper proposes an approach to water flow method modification for text segmentation and reference text line detection of sample text at almost any skew angle. Original water flow algorithm assumes hypothetical water flows under only a few specified angles of the document image frame from left to right and vice versa. As a result of water flow algorithm, unwetted image frames are extracted. These areas are of major importance for text line parameters extraction as well as for text segmentation. Water flow method modification means extension values of water flow specified angle and unwetted image frames function enlargement. Modified method is examined and evaluated under different sample text skew angles. Results are encouraged especially due to improving text segmentation which is the most challenging process stage.

Keywords: Document image processing, Reference text line, Text line segmentation, Water flow algorithm

1 Introduction

Previous work on text parameter detection can be categorized in few types:

- Histogram analysis,
- Docstrum (k-nearest neighbor clustering),
- Projection profile (Hough transform),
- Fourier transform,
- Cross-correlation,
- Other methods.

In [1] is mentioned previously proposed and accepted technique of reference line extraction based on identifying valleys of horizontal pixel density histogram. Method failed due to multi-skewed text lines.

The Docstrum method [2] is by product of a larger page layout analysis system, which assumed that only text is being processed. The connected components formed by the nearest neighbors clustering are essentially characters only. The method is suitable for finding skew angle. But, it is limited to Roman languages due to poor text line segmentation.

Another method proposed in [2, 3] deal with "simple" multi-skewed text. It uses as a basis simple type of Hough transform for straight lines. But, it's too specific.

The Fourier transform method is a representation in the Fourier domain of the projection profile method in the pixel domain. The results are mathematically identical, but Fourier transform is only different approach to the same text and document properties that projection profile is based upon [2].

The cross-correlation method calculates both horizontal and vertical projection profiles and then compares the shift inter-line cross-correlation to determine the skew rate. Although method can handle complex layout structure documents, applied range is limited to (-10°, 10°) [2].

Algorithm proposed by [4] model text line detection as an image segmentation problem by enhancing text line structure using a Gaussian window and adopting the level set method to evolve text line boundaries. Author specified method as robust, but rotating text by an angle of 10° has an impact on reference line hit rate.

Method of identifying words contour area as a start of detecting baseline point proposed in [5]. The assumptions made on the word elements definition are too specific.

Method [1] hypothetically assumed a flow of water in a particular direction across image frame in a way that it faces obstruction from the characters of the text lines. This method is adopted in [6, 7], but referent line hit rate is robust for rotating origin text from ±20° around x-axis In our paper, "water flow" method [1, 6, 7] is further adopted, implemented and examined on more complex text examples.

This paper is organized as follows: In section 2 contribution to technological innovation i.e. modification of water flow algorithm is presented. Section 3 includes definition of experiment under investigation. In section 4 experimental results are presented. Results are analyzed, examined, elaborated and discussed as well. In section 5 conclusions are made.

2 Contribution to Technological Innovation

Document text image identification procedure consists of three main stages. It is shown in Fig. 1.

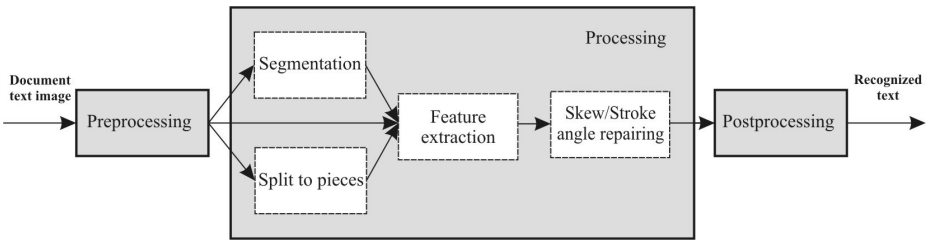


Fig. 1. Document text image identification procedure.

In preprocessing stage, algorithm for document text image binarization and normalization is applied. Now, preprocessing text is prepared for segmentation, feature

extraction and character recognition. During the processing stage, algorithms for text segmentation as well as for skew and reference text line identification are enforced. After that, reference text based on skew and stroke angle, is straightened and repaired. Finally, in postprocessing stage character recognition process is applied.

In this paper, some elements of preprocessing and processing stages are employed. A few assumptions should be made before defining algorithm. We suppose that there is an element of preprocessing. After preprocessing, document text image is prepared for segmentation and feature extraction. So, it represents distinct entity consists of group of words.

Document text image is an input of text grayscale image described by following intensity function:

$$I(l,k) \in [0,...,255] \quad , \tag{1}$$

where $l \in [0, N-1]$ and $k \in [0, M-1]$.

After applying intensity segmentation with binarization, intensity function is converted into binary intensity function given by:

$$I_{bin}(l,k) = \begin{cases} 1 & for \quad I(l,k) \geq I_{th} \\ 0 & for \quad I(l,k) < I_{th} \end{cases} \quad , \tag{2}$$

where I_{th} is given by Otsu algorithm [8].

Now, text lines are represented as digitized document image by matrix **X** with M x N dimension. Each word in document image consists of black points i.e. pixels. Every point is represented by number of coordinate pairs such as:

$$X(i,j) \in [0,1] \quad , \tag{3}$$

where $i = 1, \ldots, N$, $j = 1, \ldots, M$ of the matrix **X** [9, 10].

Original water flow algorithm assumes hypothetical water flows under only few specified angles of the document image frame from left to right and vice versa [1]. But, previously it needs to define pixel type from situation in Fig. 2.

	i	i+1
j-1	o	o
j	x	x
j+1	x	o

(a)

	i	i+1
j-1	o	o
j	x	o
j+1	x	o

(b)

	i	i+1
j-1	o	o
j	x	o
j+1	x	o

(c)

Fig. 2. Pixel type determination: (a) Upper boundary pixel, (b) Lower boundary pixel, and (c) Boundary pixel for additional investigation (x represents black and o represents white pixel).

Proposed algorithm verifies boundary pixel type in document image. After verification it makes unwetted areas around the words. Due to pixel type, i.e. upper or lower, slope is α or $-\alpha$. Additional investigation is made on pixel without complete location. It can be lower, upper or no boundary pixel. It depends on neighbor area of

pixels. Apart from [8] and [9] enlarged window R x S pixels is defined as a basis. For analysis, it is proposed R = 5 and S = 7 [7]. Position of window is backwards from pixel candidate for additional investigation. After additional investigation pixel type is completely located [7]. Throughout previous decision making, unwetted areas algorithm simply draws area under specified angles. As a result words are bounded by unwetted dark stripes. These regions are mark out by lines defined as function:

$$y = k \cdot x \ , \tag{4}$$

where slope $k = tan(\alpha)$. Lines defined by slope make connection in specific pixel creating unwetted area defined as grey region in Fig. 3.

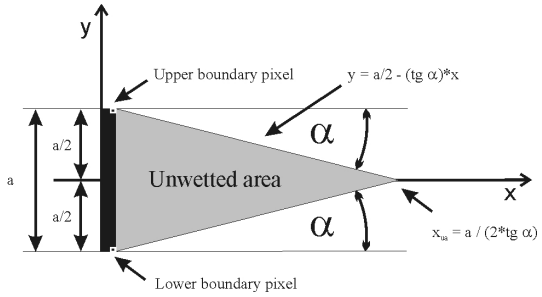


Fig. 3. Unwetted area definition.

Basic water flow algorithm is proposed with water flow fixed angles of: 45º, 26.6º, 18.4º and 14º by masking out original document image [1]. First modification made on water flow algorithm is its extension in formulation of algorithm. Still, making straight lines from boundary pixel type and connecting each others in specified point makes unweted region as well. Hence, modified water flow algorithm is free to choose different α from 0º to 90º. Unfortunately, whole range of α isn't applicable due to impossibility of joining words to form text line regions.

Further improvement is made on defining water flow function from (4). From mathematical calculus it is known that (4) is special case of more universal function:

$$y = k \cdot x^n \ , \tag{5}$$

where (4) and (5) are equal for $n = 1$.

Using different n values from (5) rebuild different shape of unwetted areas around the words. The main achieving of unwetted area is to be exploited for text segmentation. The problem lies in broken words in text lines. Algorithm should join those words by unwetted areas. Unwetted areas can lengthen by using smaller angle α or different water flow function. Hence, another modification of algorithm function means more slow down of slope forming unwetted area. Expanded unwetted area is given in Fig. 4.

Water flow function is changed from (4) to (5) with power value $0 < n < 1$.

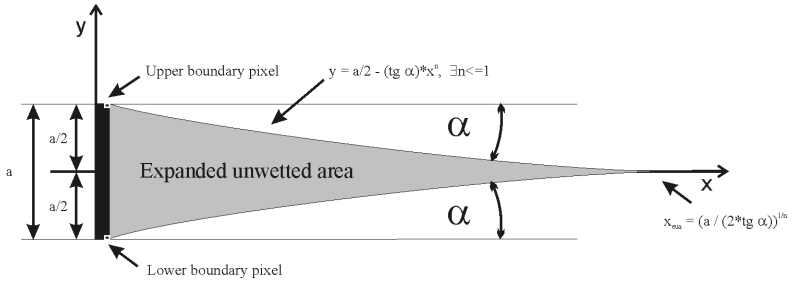


Fig. 4. Expanded unwetted area definition.

Basic and expanded water flow function zeroes are marked as x_{ua} and x_{eua}, respectively. Their difference $x_{diff} = x_{eua} - x_{ua}$ is given by:

$$x_{diff} = \frac{a}{(2 \cdot \tan \alpha)^{\frac{1}{n-1}}} \quad . \tag{6}$$

For parameter $n < 1$, x_{diff} value is greater than 0. Thus, unwetted area is expanded. The skew angle detection is based on information obtained from presented algorithm. Defining reference text line means calculating specific average position of every column of document image. It is average position of only black pixels in every column of document image. Relation for calculating reference text line is given by [1]:

$$x_i = \frac{\sum_{j=1}^{L} y_j}{L} \quad _{i=1,\ldots,K} \quad , \tag{7}$$

where x_i is calculated point represent reference text line point, i is specified column, y_j is position of black pixel in column j and L is sum of black pixel in specified column j of the document image.

3 Experiment

For the experiment, sample text rotated from -45° to +45° by step of 5° around x-axis is used. Sample text is given in Fig. 5. Sample text reference line is represented by:

$$y = a \cdot x + b \quad . \tag{8}$$

After applying algorithm, calculated document image from (7), with only one black pixel per column, is obtained. That black pixel per column defines calculated reference text line and text line skewness. Calculated reference text line forms continuous or discontinuous line partly or completely representing reference text line.

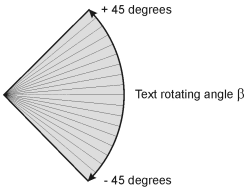


Fig. 5. Sample text rotating from –45° to +45° for the algorithm robustness investigation.

To achieve continuous linear aspect of reference text line from point's collection, least square method is used. To find a first degree polynomial function approximation given by:

$$y = a' \cdot x + b' \quad , \tag{9}$$

number of data points is used as *ndp* and the slope *a'*, and the y-intercept *b'* are calculated as [10]:

$$a' = \frac{(\sum y) \cdot (\sum x \cdot y) - ndp \cdot (\sum x \cdot y)}{(\sum x)^2 - ndp \cdot (\sum x^2)} \quad , \tag{10}$$

and

$$b' = \frac{(\sum x) \cdot (\sum x \cdot y) - (\sum y) \cdot (\sum x^2)}{(\sum x)^2 - ndp \cdot (\sum x^2)} \quad . \tag{11}$$

Further, referent line hit rate abbreviated by *RLHR* is defined by:

$$RLHR = 1 - \frac{\beta_{ref} - \beta_{est}}{\beta_{ref}} \quad , \tag{12}$$

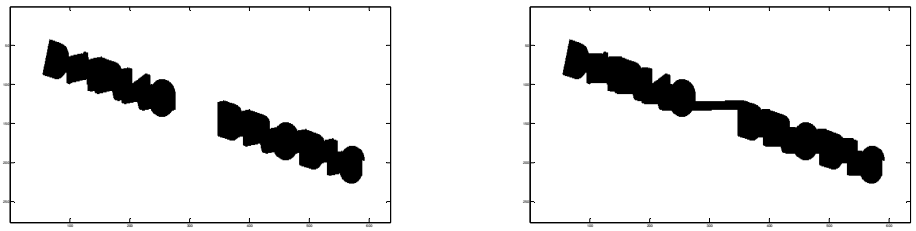
where β_{ref} is arctangent of *a* (origin) from (8) and β_{est} is arctangent of *a'* (calculated i.e. estimated) from (9). *RMS* values are calculated by [10]:

$$RMS = \sqrt{\frac{1}{R} \sum_{i=1}^{R} (x_{ref} - x_{est})^2} \quad , \tag{13}$$

where *R* is number of examined text rotating angles range from ±60°, x_{ref} is *RLHR* for β_{est} equal to β_{ref} i.e. due to normalization equal to 1, and x_{est} is *RLHR*.

4 Results and Discussion

Benefit from extended algorithm is perceived in text segmentation process. Split up words in text line is presumably joined by unwetted areas from water flow function. Two word connections made by basic and extended algorithm are given in Fig. 6.



(a) (b)

Fig. 6. Document text image with unwetted areas: (a) basic algorithm, and (b) extended algorithm (power $n = ½$ from (5)).

Sample text *RLHR* for basic and extended algorithm is given in Fig. 7.

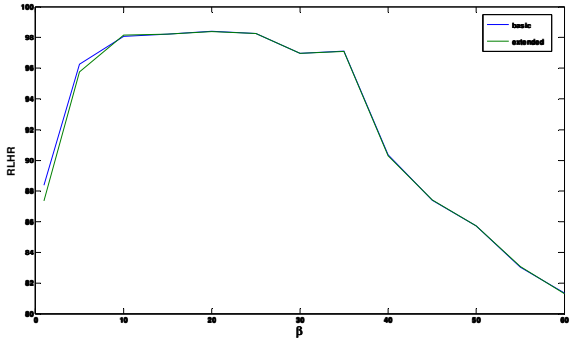


Fig. 7. Sample text *RHLR* (α from 10° to 30° by step 5°, β from 5° to 60° by step 5°): basic vs. extended algorithm ($n = ½$).

RLHR for basic and extended algorithm are almost identical as can be seen from Fig. 7. For sample text rotating up to 40° *RLHR* is at least 90% or better. It proved robustness of both algorithms. *RLHR* isn't reduced due to better segmentation by extended algorithm. Sample text *RMS* for basic and extended algorithm is given in Fig. 8.

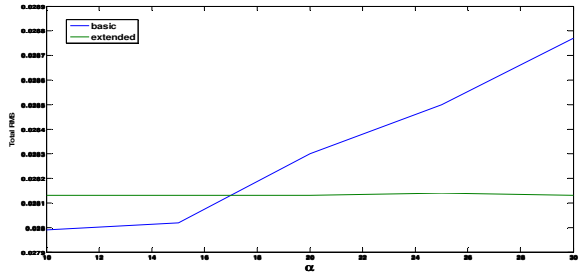


Fig. 8. Sample text *RMS* (α from 10° to 30° by step 5°, β from 5° to 60° by step 5°): basic algorithm vs. extended algorithm ($n = ½$**).**

RMS made by basic algorithm for α from 10° to 30° is growing on. Unlike, *RMS* made by extended algorithm is steady. Due to *RMS* condition, magnitude of varying quantities is similar for the whole investigated region of the unwetted angles in extended algorithm.

5 Conclusion

In this paper an approach to modified water flow algorithm for text segmentation and reference text line extraction is presented. Water flow algorithm assumes hypothetical water flows under few specified angles of the image frame from left to right and vice versa. As a result of algorithm, unwetted image regions defined by function $y = kx$ are restored. Modified water flow algorithm made water to flow under whole range of different angles. Same way, it exchanges function defining unwetted image regions to $y = kx^n$. Those unwetted regions are corner stone needed for reference text line calculation. Using least square method on calculated reference text line, reference text line is defined and extracted. Modified water flow function proved better text segmentation preferences. Robustness of algorithm is examined and validated using sample text rotating in region of ±60° around x-axis. Results confirmed good *RLHR* values for text rotating in region of ±45°. Further improvements of algorithm should be made on better *RLHR* in full region of ±90°. One way is algorithm throw out to strictly follow reference text line turn over.

References

1. Basu S., Chaudhuri C., Kundu M., Nasipuri M., Basu D.K.: Text Line Extraction from Multi-Skewed Handwritten Documents. In: Pattern Recognition, Vol.40, pp. 1825--1839 (2006)
2. Amin A., Wu S.: Robust Skew Detection in mixed Text/Graphics Documents. In: Proceedings of 8th International Conference on Document Analysis and Recognition (ICDAR '05), pp. 247--251 Seoul, Korea (2005)
3. Louloudis G., Gatos B., Pratikakis I., Halatsis C.: Text Line Detection in Handwritten Documents. In: Pattern Recognition, Vol.41, pp. 3758--3772 (2008)
4. Li Y., Zheng Y., Doermann D., Jaeger S.: A New Algorithm for Detecting Text Line in Handwritten Documents. In: Proceedings of 18th International Conference on Pattern Recognition, Vol.2, pp. 1030-1033. Hong Kong, China (2006)
5. Wang J., Mazlor K., Leung H., Hui S.C.: Cursive Word Reference Line Detection. In: Pattern Recognition, Vol.30, No.3, pp. 503--511 (1997)
6. Brodić D., Milivojević Z.: Reference Text Line Identification Based on Water Flow Algorithm. In: Proceedings of ICEST '2009, SP-2 Sect., Veliko Tarnovo, Bulgaria (2009)
7. Brodić D., Milivojević Z.: Modified Water Flow Method for Reference Text Line Detection. In: Proceedings of ICCS '2009, Sofia, Bulgaria (2009)
8. Gonzalez R.C., Woods R.E.: Digital Image Procesing, 2nd ed., New Jersey: Prentice-Hall, pp. 67--70 (2002)
9. Sonka M., Hlavac V., Boyle R.: Image Processing, Analysis and Machine Vision, Toronto: Thomson, pp. 174--177 (2008)
10. Bolstad W.M.: Introduction to Bayesian Statistics, New Jersey: John Wiley & Sons, 2004, pp. 40--44, 235--240 (2004)