Darko BRODIC

University of Belgrade, Technical Faculty Bor

Methodology for the evaluation of the algorithms for text segmentation based on errors type

Abstract. *Text segmentation represents the key element in the optical character recognition process. Hence, testing procedure for text segmentation algorithms has significance importance. All previous works deal mainly with text database as a template. They are used for testing as well as for the evaluation of the text segmentation algorithm. However, because of inconsistencies in this process, some methodology for the experiments is required. In this manuscript, methodology for the evaluation of the algorithm for text segmentation based on errors type is proposed. It is established on the various multiline text samples linked with text segmentation. Final result is obtained by comparative analysis of cross linked data. At the end, its suitability for different type of scripts represents its main advantage.*

Streszczenie. *Segmentacja tekstu stanowi kluczowy element procesu optycznego rozpoznawania znaków. Wszystkie dotychczasowe prace dotyczą głównie bazy danych tekstu jako szablonu. Są one używane do testowania, jak i dla oceny algorytmu segmentacji tekstu. Jednak w taki, algorytmie występują nieścisłości. W pracy przedstawiono , metodologię oceny algorytmu segmentacji tekstu w oparciu o typ błędów. Badania przeprowadzono na różnych próbkach tekstu wielowierszowego. Końcowy wynik uzyskuje się poprzez analizę porównawczą danych.* (***Metodologia oceny algorytmów segmentacji tekstu w oparciu o błędy typu***).

Keywords: Optical character recognition, document image processing, text segmentation, testing methodology.
Słowa kluczowe: Optyczne rozpoznawanie znaków, przetwarzanie obrazu dokumentu, segmentacja tekstu, metodologia badań.

Introduction

Text line segmentation is an important step in document image processing [1]. It represents a labeling process which consists in assigning the same label to spatially aligned units [2]. Although text line detection techniques are mainly successful in printed documents, processing of the handwritten documents has remained a key problem [3]. Consequently, text line segmentation of handwritten documents is a complex and diverse problem, complicated by the nature of handwriting [4]. Hence, it is a leading challenge in handwritten document image processing [2]. Many proposed algorithms for text line segmentation have been evaluated by quite different test methods. In fact, these evaluation procedures are usually based on the use of a custom text database as a test sample [5-6]. Accordingly, testing result interpretation is quite dissimilar [7]. Due to inconsistencies in measurement and evaluation of text segmentation algorithm quality, some basic methodology is required. Currently, there is no commonly accepted one [8]. Hence, the establishment of the methodology for the evaluation of the algorithm's text segmentation is of major importance.

In this paper, methodology for the evaluation of algorithms for text segmentation is proposed. It is based on the experiments linked with synthetic text samples as well as real handwritten ones. Although they are mutually independent, the obtained results are cross linked. At the end, its suitability for different types of scripts represents its main advantage. Hence, the paper presents an efficient evaluation methodology for text segmentation algorithms.

The paper is organized as follows: in Section 2 the test framework and basic methodology are presented. Section 3 contains the test results evaluation process. In Section 4 testing algorithm is briefly explained. In Section 5 test results are given and comparative analysis is made. Conclusions are given in Section 6. Furthermore, in Appendix the examples of text sample scripts are presented.

Basic methodology

Testing of the algorithm represents the process of the applying algorithm to the proposed text samples. Text examples represent combination of the synthetic as well as the real ones similarly as in [9]. The major test assignment is the efficiency evaluation of the algorithm for text segmentation. Methodology for the evaluation of algorithm's text segmentation consists of few text experiments. It is based on the following activities [8, 10]:

- Multi-line text segmentation test,
- Multi-line waved text segmentation test,
- Multi-line fractured text segmentation test, and
- Real handwritten text segmentation test.

The schematic test procedure is shown in Fig. 1.

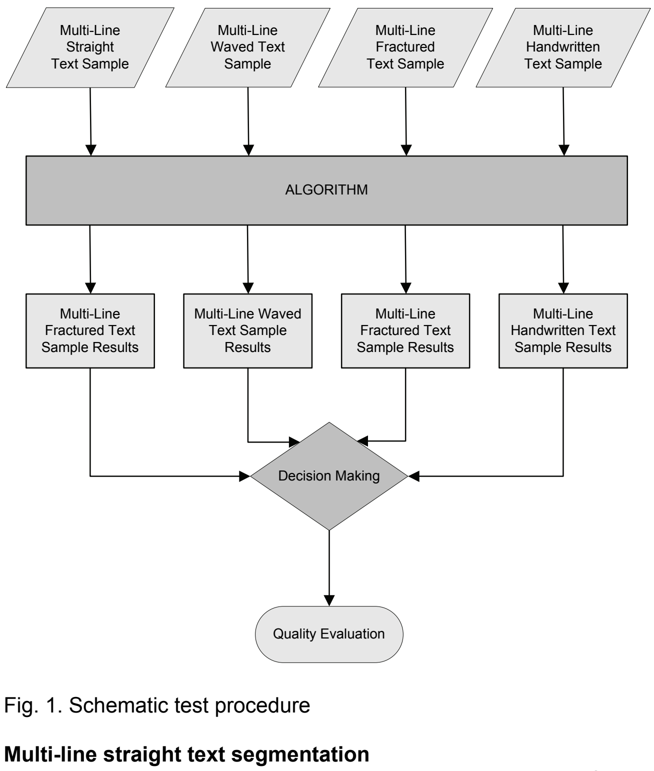
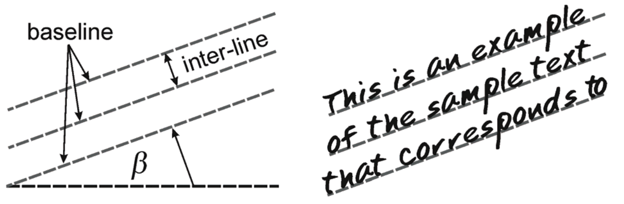


Fig. 1. Schematic test procedure

Multi-line straight text segmentation

Multi-line straight text segmentation test consists of the sample text set that is based on straight baseline. Straight text is defined by skew angle β. Typical values of β that correspond to the handwritten text are those up to 20°. Hence, it takes value from the set {5°, 10°, 15°, 20°} [8, 10]. Furthermore, inter-line spacing is set to 20% of the standard character height [11]. This corresponds to single line spacing [12]. The resolution of the text samples is 300 dpi. Each set of multi-line straight text samples incorporate 96 lines of: Latin, Cyrillic, Glagolitic and Bengali text. Multi-line straight text definition is illustrated in Fig. 2.



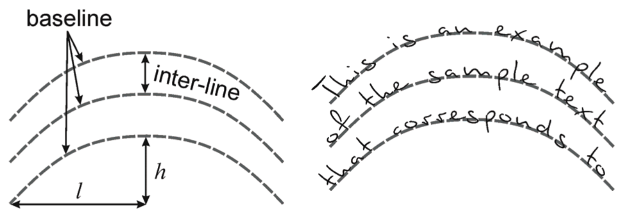
(a) (b)

Fig. 2. Multi-line straight text: (a) definition, (b) sample.

Fragments of different sample text scripts are shown in Fig. A1.

Multi-line waved text segmentation

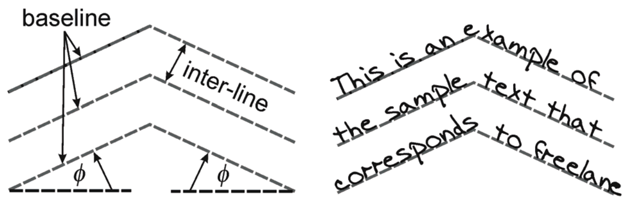
Multi-line waved text segmentation test consists of the sample text set that is based on wavy baseline. Waved text is defined by the parameter ε. It is given as $\varepsilon = h/l$, where h is height, and l is half-width of the waved baseline. Typical values of ε that correspond to the previously chosen values of skew angle β are from the set {1/12, 1/6, 1/4, 1/3} [8, 10]. Inter-line spacing is set to 20% of the standard character height [11]. The resolution of the text samples is 300 dpi. Each set of multi-line waved text samples incorporate 96 lines of: Latin, Cyrillic, Glagolitic and Bengali text. Multi-line waved text definition is illustrated in Fig. 3.



(a) (b)

Fig. 3. Multi-line waved text: (a) definition, (b) sample.

Fragments of different sample text scripts are shown in Fig. A2.



(a) (b)

Fig. 4. Multi-line fractured text: (a) definition, (b) sample.

Multi-line fractured text segmentation

Multi-line fractured text segmentation test consists of the sample text set that is based on fractured baseline. Fractured text is defined by the fractured skew angle γ. Typical values of γ that correspond to the handwritten text are those up to 20°. Hence, it receives value from the set {5°, 10°, 15°, 20°} [8, 10]. Furthermore, inter-line spacing is set to 20% of the standard character height [11]. Resolution of the text samples is 300 dpi. Each set of multi-line fractured text samples incorporate 96 lines of: Latin, Cyrillic, Glagolitic and Bengali text. Multi-line fractured text definition is illustrated in Fig. 3.

Fragments of different sample text scripts are shown in Fig. A3.

Multi-line handwritten text segmentation

Multi-line handwritten text segmentation test is based on freestyle handwritten text samples in Serbian Latin, Cyrillic as well as in English script [13]. This is a document image text database which total number of handwritten text samples consists of 220 text lines. These text samples contain variable skew lines i.e. multi-oriented text. Resolution of the text samples is 300 dpi. Few handwritten text fragments from the text database are shown in Fig. A4.

Test results evaluation

For the measurement evaluation of text line segmentation following terms are defined [11, 13]:

- Initial objects,
- Detected objects, and
- Referent objects.

Initial objects O_{in} represent the starting number of objects in referent sample text. After applying the algorithm over referent sample text counted number of objects is given as detected objects O_{dt}. Further, the goal, i.e. desired number of objects O_{ref} represents the number of text lines in referent sample text. It is called referent number of objects. By comparing the referent and detected number of objects per each line the algorithm efficiency is evaluated.

Classification of the text segmentation errors

If the number of text objects in distinct text line is equal to one, then $O_{dt} = O_{ref}$ leading to correctly segmented text line. The number of correctly detected text lines in sample text is marked as $O_{corlindet}$. However, all others are defined as error. These circumstances are illustrated in Fig. 5.

Segmentation errors are present in the following circumstances:

- Split lines error (*SLE*) [14] i.e. over-segmentation detected text lines $O_{overlindet}$, and
- Joined lines error (*JLE*) [14] i.e. under-segmentation detected text lines $O_{underlindet}$.

Split lines error represents the text lines which are wrongly divided by algorithm in two or more components, i.e. text objects [14]. This circumstance is known as over-segmentation. Joined lines error corresponds to the situation where the sequence of n consecutive lines is considered by the algorithm as a unique line [14]. This phenomenon is called under-segmentation.

Algorithms evaluation based on errors type

The algorithms efficiency means the evaluation of the text segmentation process made by investigated algorithm. If the number of detected objects is closer to the number of referent objects, then the algorithm is more efficient. To evaluate the algorithm's efficiency the following elements are introduced:

- Segmentation line hit rate, i.e. *SLHR*,
- Over-segmentation line hit rate, i.e. *OSLHR*,
- Under-segmentation line hit rate, i.e. *USLHR*, and
- Segmentation root mean square error $RMSE_{seg}$.

SLHR represents the ratio of the number of correctly segmented text lines over the total number of text lines in the referent sample text. It is defined as:

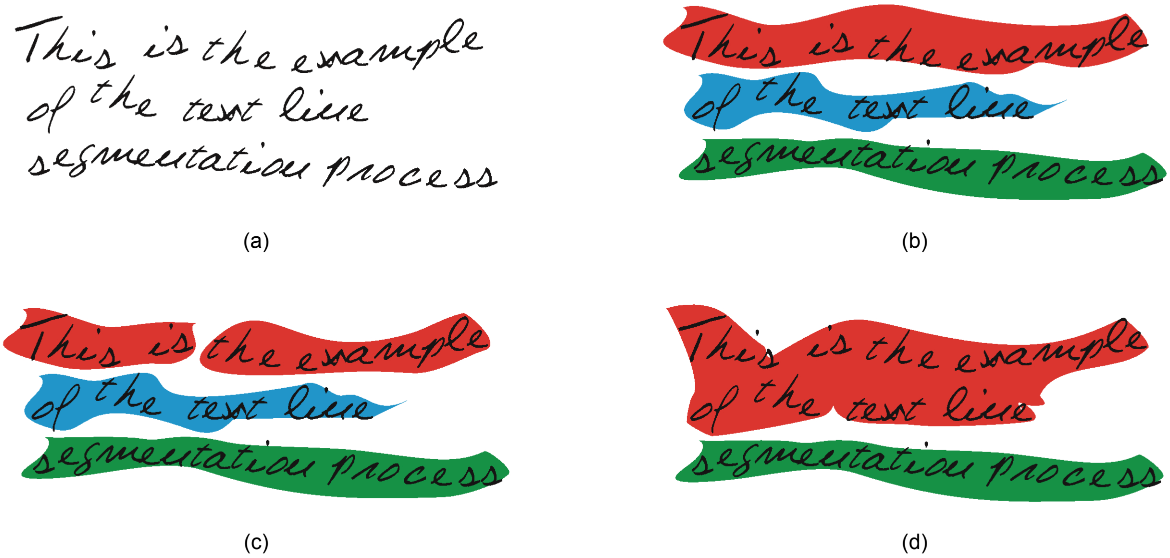


Fig. 5. Text line segmentation: (a) Initial text, (b) Correctly segmented text lines, (c) Over-segmentation text lines (line #1 over-segmented), (d) Under-segmentation text lines (lines #1 and #2 connected, i.e. under-segmented).

(1) $$SLHR = 1 - \left| \frac{O_{ref} - O_{corlindet}}{O_{ref}} \right| .$$

The over-segmentation phenomena lead to the increased number of objects per text line. Hence, boundary growing area made by algorithm hasn't been successful in merging all objects of the text line into one. As previously stated, the number of the over-segmented lines is marked as $O_{overlindet}$. *OSHLR* represents the ratio of the number of over-segmented text lines over the total number of text lines in the referent sample text. It is defined as:

(2) $$OSLHR = 1 - \left| \frac{O_{ref} - O_{overlindet}}{O_{ref}} \right| .$$

The under-segmentation process leads to the smaller number of objects than the number of text lines. Hence, two or more consecutive text lines are considered as a unique one. *USHLR* represents the ratio of the number of under-segmented text lines over the total number of text lines in the referent sample text. It is defined as:

(3) $$USLHR = 1 - \left| \frac{O_{ref} - O_{underlindet}}{O_{ref}} \right|$$

At the end, the number of detected and referent text objects per each text line are compared. However, the number of referent text objects per line is equal to 1. The variance evaluation is given $RMSE_{seg}$ [8, 10]:

(4) $$RMSE_{seg} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(O_{i,ref} - O_{i.est}\right)^2}$$

where N is the total number of lines in the sample text, $O_{i,ref}$ is the number of referent objects in the text line i (equal to 1 per each line), and $O_{i,est}$ is the number of detected objects in the text line i.

Testing algorithm

For the testing purposes, smearing method sample for text segmentation is used. It represents the group of boundary growing algorithms. In smearing methods the consecutive black pixels along the horizontal direction are smeared [15]. The seed points that fulfill predefined criteria activate process. Consequently, the white space between black pixels is filled with black pixels. It is achieved only if their distance is within a predefined threshold. This way, enlarged areas of black pixels around text are formed. It is so-called boundary growing areas. These areas of the smeared image enclose separated text lines. Hence, obtained areas are mandatory for text line segmentation.

As a smearing method, the Gaussian anisotropic kernel algorithm is used [5]. This algorithm will be just briefly explained for the purpose of testing. Its main task is expanding black pixel areas of text by scattering every black pixel in its neighborhood. This way, distinct areas that mutually separate text lines are established. Its primary purpose is joining only text elements from the same text line into the same distinct continuous areas. Gaussian probability function is taken as template that gives the probability of the random function. Consequently, it represents probability of the hypothetical expansion around every black pixel that represents a text element. Hence, around every black pixel, new pixels are non-uniformly dispersed. These new pixels have lower black intensity. Because the level of probability expansion relates to distance from black pixel, their intensity depends completely in regard to the distance from the original black pixel. Newly formed pixels are grayscale. Hence, document text image is a grayscale. However, after applying Gaussian anisotropic kernel, equal to $2K+1$ in x-direction and $2L+1$ in y-direction, text is scattered forming an enlarged area around it. Now, inside the kernel a "probability" sub area is formed using the radius $3\sigma_x$ and $3\sigma_y$ of ellipse in x and y direction. Consequently, σ represents standard deviation defining curve spread parameter. Converting all these pixels into black pixels as well as inverting image, forms the new black pixel expanded areas [16-17]. These areas are named boundary-growing areas.

Results and comparative analysis

The main purpose of the testing is the optimization of the algorithm parameters. In our example, parameters of interest are *K* and *L*. They define kernel size. Testing text samples incorporate letters which height is up to 60 pixels. In compliance with it *K* should be chosen approximately from 10% to 20% of the letters height [16]. Obtained results for different test sets are presented in Tables 1-4.

From table 1, the best results concerning *SLHR* are those obtained using parameters pair (*K*, *L*) from the following set: (8, 32), (8, 40), and (10, 40). Obviously, using higher *K* (i.e. equal to 10) leads to better *SHLR* results for waved and fractured text. However, the results for straight text are disappointing. On contrary, the choice of parameters pair equal to (8, 40) brings uniform *SLHR* values. Further, to overcome over-segmentation problem the choice of pair (10, 40) is correct one. From above results using higher *L* value leads to better text line segmentation. However, it raises the under-segmentation phenomena as well. Hence, careful decision making is needed to overcome it. At the end, $RMSE_{seg}$ confirms results from tables 1-3. However, to improve the behavior of the testing algorithm additional algorithm for the evaluation of between line distance is prerequisite. Incorporation of this algorithm will reduce the under-segmentation phenomena leading to better text line segmentation results evident by higher *SLHR* and smaller $RMSE_{seg}$ values.

Finally, for the text samples that includes letters which height is up to 60 pixels, the optimal kernel size is 2*K*+1 x 2*L*+1, i.e. (2x8)+1 x (2x40)+1 px. This leads to $K \approx 15\%$ letter height [16] and $L \approx 5 \times K$.

Conclusions

This paper describes proposal of the methodology for the evaluation of the algorithms text line segmentation. All previous evaluation procedures were custom oriented. However, the proposed test method is a step toward generalization of the test procedure in the domain of document image processing algorithms evaluation. It consists of the test experiments that measure text line segmentation algorithm ability. They incorporate four various multi-line text experiments: straight, waved, fractured and handwritten ones. Each of them consists of different script types. For testing purposes, Gaussian anisotropic kernel algorithm is used. After obtaining measurement results validation of the algorithm is performed. Hence, suitable validation method based on text segmentation errors is proposed. It is classified by distinct measures *SLHR*, *OSLHR*, *USLHR*, and $RMSE_{seg}$. However, they are inter-related as well. They are combined after decision making by the intersection of obtained results. The benefit of this process is optimized subset of algorithm parameters values. This process is invaluable for algorithm evaluation. At the end, its suitability for different types of letters and languages as well as its adaptability is a strong advantage.

Appendix

In appendix various fragments of different sample text scripts are given. These text samples incorporate Latin, Cyrillic, Glagolitic and Bengali scripts. They are given in figures A1-A4.

REFERENCES

[1] Du X., Pan W., Bui T.D., Text line segmentation in handwritten documents using Mumford–Shah model, *Pattern Recognition*, 42(2009), No.12, 3136-3145.

[2] Likforman Sulem L., Zahour A., Taconet B., Text line segmentation of historical documents: a survey, *International Journal on Document Analysis and Recognition*, 9(2007), No.2, 123-138.

[3] Amin A., Wu S., Robust skew detection in mixed text/graphics documents, *Proceedings of International Conference on Document Analysis and Recognition - ICDAR*'05, Seoul, Korea, 2005, pp.247-251.

Table 1. *SLHR* (%) for different text samples (*K* and *L* given in pixels).

K	5					8					10				
L	5	10	15	20	25	8	16	24	32	40	10	20	30	40	50
Straight text	0.00	0.00	81.25	91.67	95.83	0.00	83.33	95.83	85.42	72.92	0.00	87.50	81.25	64.58	58.33
Waved Text	0.00	0.00	0.00	0.00	6.25	0.00	0.00	6.25	62.50	95.83	0.00	0.00	58.33	100.00	100.00
Fractured Text	0.00	0.00	0.00	0.00	0.00	0.00	0.00	6.25	75.00	87.50	0.00	0.00	56.25	83.33	81.25

Table 2. *OSLHR* (%) for different text samples (*K* and *L* given in pixels).

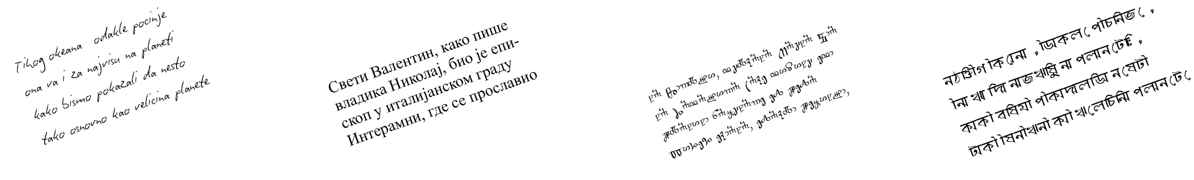
K	5					8					10				
L	5	10	15	20	25	8	16	24	32	40	10	20	30	40	50
Straight text	100.00	100.00	18.75	6.25	2.08	100.00	14.58	0.00	0.00	0.00	89.58	0.00	0.00	0.00	0.00
Waved Text	100.00	100.00	100.00	100.00	93.75	100.00	100.00	93.75	37.50	4.17	100.00	100.00	41.67	0.00	0.00
Fractured Text	100.00	100.00	97.92	95.83	95.83	97.92	95.83	89.58	16.67	0.00	91.67	91.67	33.33	0.00	0.00

Table 3. *USLHR* (%) for different text samples (*K* and *L* given in pixels).

K	5					8					10				
L	5	10	15	20	25	8	16	24	32	40	10	20	30	40	50
Straight text	0.00	0.00	0.00	2.08	2.08	0.00	2.08	4.17	14.58	27.08	10.42	12.50	18.75	35.42	41.67
Waved Text	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Fractured Text	0.00	0.00	2.08	4.17	4.17	2.08	4.17	4.17	8.33	12.50	8.33	8.33	10.42	16.67	18.75

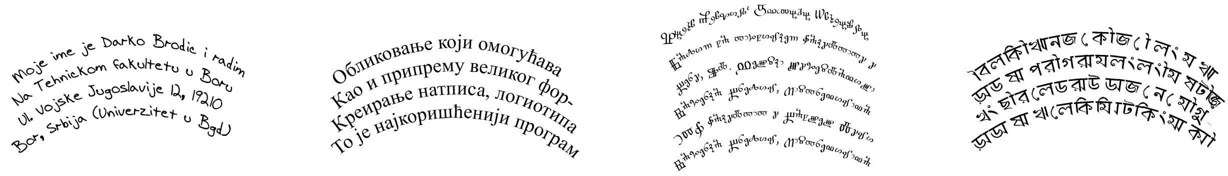
Table 4. $RMSE_{seg}$ for different text samples (*K* and *L* given in pixels).

K	5					8					10				
L	5	10	15	20	25	8	16	24	32	40	10	20	30	40	50
Straight text	3.30	3.06	0.61	0.29	0.20	3.19	0.41	0.20	0.38	0.52	2.90	0.35	0.43	0.60	0.65
Waved Text	9.75	4.73	3.49	3.11	2.46	4.63	3.11	2.61	0.66	0.20	3.77	3.08	0.85	0.00	0.00
Fractured Text	7.70	4.23	4.07	4.01	3.18	4.01	3.86	3.42	0.61	0.35	3.71	3.69	1.34	0.41	0.43



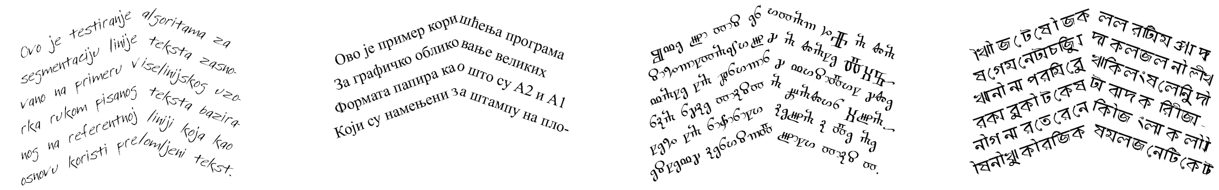
(a) (b) (c) (d)

Fig. A1. Multi-line straight text: (a) Latin text, (b) Serbian Cyrillic text, (c) Glagolitic text , (d) Bengali text.



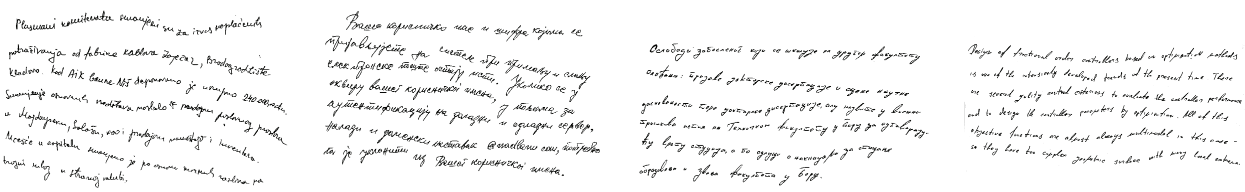
(a) (b) (c) (d)

Fig. A2. Multi-line waved text: (a) Latin text, (b) Serbian Cyrillic text, (c) Glagolitic text , (d) Bengali text.



(a) (b) (c) (d)

Fig. A3. Multi-line fractured text: (a) Latin text, (b) Serbian Cyrillic text, (c) Glagolitic text , (d) Bengali text.



(a) (b) (c) (d)

Fig. A4. Multi-line handwritten text fragments: (a) Serbian Latin text, (b) Serbian Cyrillic text, (c) Cyrilic text, (d) English text.

[4] Bukhari S.S., Shafait F., Breuel T.M., Script-Independent handwritten textlines segmentation using active contours, *Proceedings of International Conference on Document Analysis and Recognition - ICDAR*'09, Barcelona, Spain, 2009, pp.446-450.

[5] Yi L., Yefeng Z., Doermann D., Jaeger S., Script-Independent text line segmentation in freestyle handwritten documents, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2008), No.8, 1313-1329.

[6] Basu S., Chaudhuri C., Kundu M., et. al., Text line extraction from multi-skewed handwritten document, *Pattern Recognition*, 40(2007), No.6, 1825-1839.

[7] Razak Z., Zulkiflee K., et al., Off-Line handwriting text line segmentation: a review, *International Journal of Computer Science and Network Security*, 8(2008), 12-20.

[8] Brodić D., Milivojević D.R., Milivojević Z., Basic test framework for the evaluation of text line segmentation and text parameter extraction, *Sensors* 10(2010), No.5, 5263-5279.

[9] Khashman A., Sekeroglu B., Document image binarisation using a supervised neural network, *International Journal of Neural Systems* 18(2008), No.5, 405-418.

[10] Brodić D., Basic experiments set for the evaluation of the text line segmentation, *Przeglad Elektrotechniczny* (*Electrical Review*), 86(2010), No.11, 353-357.

[11] Brodić D., Milivojević Z., Milivojević D.R., Improvement of the Text Line Segmentation by Rotating Anisotropic Gaussian Kernel, *Elektronika ir Elektrotechnika* (*Electronics and Electrical Engineering*) (submitted - under review).

[12] Zramdini A., Inglod R., Optical font recognition from projection profiles, *Electronic Publishing*, 6(1993), No.3, 249-260.

[13] Brodić D., Advantages of the extended water flow algorithm for handwritten text line segmentation, *Proceedings of the 4th International Conference Pattern Recognition and Machine Intelligence - PReMI* '11, Kuznetsov S.O. et al. (Eds.), LNCS 6744, Moscow, Russia, 2011, pp.418-423.

[14] Sanchez A., Suarez P.D., Mello C.A.B., et al., Text line segmentation in images of handwritten historical documents, *Proceedings of First International Workshops on Image Processing Theory, Tools and Applications*, Sousse, Tunisia, 2008, pp.1–6.

[15] Shi Z., Govindaraju V., Line separation for complex document images using fuzzy runlength, *Proceedings of the International Workshop on Document Image Analysis for Libraries*, Palo Alto, U.S.A., 2004.

[16] Brodić D., Optimization of the anisotropic Gaussian kernel for text segmentation and parameter extraction., *Proceedings of Theoretical Computer Science - TCS*'2010, Calude C.S. & Sassone V. (Eds.), IFIP AICT 323, Brisbane, Australia, 2010, pp.140-152.

[17] Brodić D., Milivojević Z., Optimization of the Gaussian kernel extended by binary morphology for text line segmentation, *Radioengineering*, 19(2010), No.4, 718-724.

***Author**: Res. assistant Darko Brodic, University of Belgrade, Technical Faculty Bor, ul. Vojske Jugoslavije 12, 19210 Bor, Serbia, E-mail: dbrodic@tf.bor.ac.rs.*