

Methodology for the Evaluation of the Algorithms for Text Line Segmentation Based on Extended Binary Classification

D. Brodic

University of Belgrade, Technical Faculty Bor, V. J. 12, 19210, Bor, Serbia, e-mail: dbrodic@tf.bor.ac.rs

Text line segmentation represents the key element in the optical character recognition process. Hence, testing of text line segmentation algorithms has substantial relevance. All previously proposed testing methods deal mainly with text database as a template. They are used for testing as well as for the evaluation of the text segmentation algorithm. In this manuscript, methodology for the evaluation of the algorithm for text segmentation based on extended binary classification is proposed. It is established on the various multiline text samples linked with text segmentation. Their results are distributed according to binary classification. Final result is obtained by comparative analysis of cross linked data. At the end, its suitability for different types of scripts represents its main advantage.

Keywords: Document image processing, text line segmentation, testing methodology.

1. INTRODUCTION

TEXT LINE segmentation is an important step in document image processing [1]. It represents mainly a labeling process, which consists in assigning the same label to spatially aligned units [2]. Text line detection techniques are successful mainly in printed documents. However, processing of the handwritten documents has remained a key problem [3]. Text line segmentation of handwritten documents is a complex and diverse problem, complicated by the nature of handwriting [4]. Hence, it represents a leading challenge in handwritten document image processing [2].

Many proposed algorithms for text line segmentation used a custom text database as a sample for the test procedure [5-6]. However, they are mostly based on English document handwritten text [7] or extended by additional scripts [8]. Additionally, most of the evaluation procedures are based on a pixel-based method [9-10]. Their performance evaluation technique is based on comparison of the detected segmentation result with an already annotated ground truth [9-10]. In addition, this is an automated approach. However, the performance evaluation is a goal-oriented task. This is particularly correct for text line segmentation. Consequently, text line segmentation goal is object-line oriented scanning. Few methodologies are established on this premise [11]. Hence, similar methodology for the evaluation of algorithms for text segmentation is proposed. It is based on the experiments linked with synthetic text samples [12-13] extended by real handwritten ones [14]. They are highly adaptable and can be extended with different types of scripts and languages. Obtained measurement results are classified according to the signal detection theory. Furthermore, they are extended with additional measure.

The paper is organized as follows: in Section 2 test framework is presented. Section 3 contains the test evaluation procedure. Section 4 gives a brief introduction to the testing algorithm. In Section 5 results are analyzed and discussed as well. Conclusions are given in Section 6. Furthermore, Appendix incorporates the examples of text

sample scripts as well as the explanation of $RMSE_{seg}$ measure.

2. BASIC METHODOLOGY

Testing process represents the procedure of applying algorithm to the proposed text samples. Consequently, they consist of the synthetic as well as the real ones, similarly as in [15]. Testing assignment is to evaluate the algorithm for text segmentation. Methodology for this assignment consists of the following text experiments [12-13]:

- Multi-line straight text segmentation test,
- Multi-line waved text segmentation test,
- Multi-line fractured text segmentation test, and
- Real handwritten text segmentation test [14].

The overall block diagram of the proposed methodology is shown in Fig. 1.

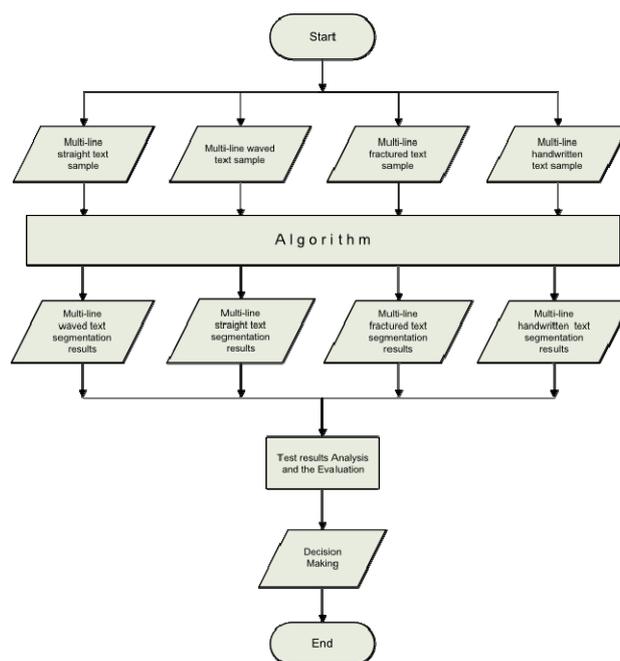


Fig.1. Schematic procedure of proposed methodology.

To evaluate the algorithm correctly, the text experiment should incorporate various classes of text types as well as scripts. Furthermore, many specific text line phenomena have to be included. These phenomena can be represented by:

- Touching connected components and text lines,
- Mixed text lines, and
- Indentation text lines (from left or right side).

Moreover, due to different baselines, which are linked indirectly with the segmentation process, different scripts are included as well. However, the majority of the above phenomena are linked with the handwritten text segmentation test.

Although test experiments are diverse, their results are inter-related. Hence, decision-making is required to combine results and make conclusions. As a result of the decision-making procedure, the set of the algorithm's parameter values is revealed. This set is the starting point for the procedure of the algorithm's optimal parameter selection.

2.1. MULTI-LINE STRAIGHT TEXT SEGMENTATION

Multi-line straight text segmentation test consists of the text samples that are based on straight baseline. However, straight baseline is defined by the parameter called skew angle β . Typical values of β that correspond to the real handwritten text are those up to 20° . For the test purposes, it is assumed that it takes value from 5° to 20° with consecutive step of 5° [12-13]. For the subsequent parameter, inter-line spacing is chosen. It is set to 20% of the standard character height [16], which corresponds to the single line spacing [17]. Accordingly, resolution of 300 dpi for text samples is used. The set of multi-line straight text samples consists of 96 lines of: Latin, Cyrillic, Glagolitic and Bengali text. Its definition is illustrated in Fig.2.

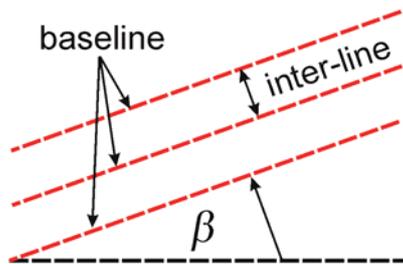


Fig.2. Multi-line straight text definition.

2.2. MULTI-LINE WAVED TEXT SEGMENTATION

Multi-line waved text segmentation test consists of the text samples that are based on waved baseline. Waving of text samples is defined by the parameter ε which is given as the ratio of waved baseline height h and half-width l , i.e., $\varepsilon = h/l$. For test purposes, typical values of ε are chosen accordingly to previously chosen values of skew angle β , i.e., from set $\{1/12, 1/6, 1/4, 1/3\}$ [12-13]. Similarly, inter-line spacing is set to 20% of the standard character height [16]. The resolution of the text samples is 300 dpi. The set of multi-line straight text samples consists of 96 lines of:

Latin, Cyrillic, Glagolitic and Bengali text. Its definition is illustrated in Fig.3.

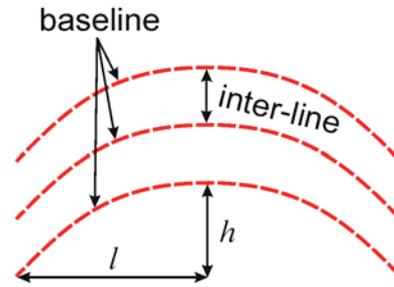


Fig.3. Multi-line waved text definition.

2.3. MULTI-LINE FRACTURED TEXT SEGMENTATION

Multi-line fractured text segmentation test consists of the text samples that are based on fractured baseline. Fractured baseline is defined by the parameter called fractured skew angle γ . Typical values of γ that correspond to the handwritten text are those up to 20° . Hence, it is assumed that it takes value from 5° to 20° with consecutive step of 5° [12-13]. Furthermore, inter-line spacing is set to 20% of the standard character height [16]. The resolution of the text samples is 300 dpi. The set of multi-line straight text samples consists of 96 lines of: Latin, Cyrillic, Glagolitic and Bengali text. Its definition is illustrated in Fig.4.

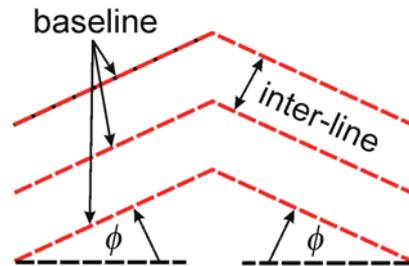


Fig.4. Multi-line fractured text definition.

2.4. MULTI-LINE HANDWRITTEN TEXT SEGMENTATION

Multi-line handwritten text segmentation test consist of the unconstrained handwritten text samples. They are written in Serbian Latin, Cyrillic as well as in English script [14]. This is a document text database which consists of 220 text lines of handwritten text. All handwritten text samples contain variable skew lines, i.e., multi-oriented text. Resolution of the text samples is 300 dpi. A few fragments of the handwritten dataset are shown in Fig.5.

3. TEST RESULTS EVALUATION

During the test procedure for text line segmentation, the algorithm is applied to the text samples. To evaluate the testing algorithm from the obtained results, following terms should be defined [14]:

- Initial connected components,
- Detected connected components, and
- Referent connected components.

Plasmanu komiteta suzjenja su za ovog popisa
 potvrdjenja od strane kablara zvezde, Brodogradiste
 Kladovo. Kod Atk. Sume MS deponimo je unijmo 240 osoba.
 Sumiranje omarice, matricu moralo je razdvojiti postojeci problem
 u Kladovu, Selo, kao i predjaci, ministar i invecija.
 Nacise u arhivu suzjeno je po ovom kriterij, razdvoje na
 knjizi i knj. u tihomj odob.

(a)

Pravica korjenicko mas i knjige kojima se
 pruzaju usluge za usluge i knjige kojima se
 klasifikovane knjige odaju masi. Ukoliko se u
 knjigu vanej korjenicko masi, u knjige za
 autentifikaciju, na razdvoje u knjige, razdvoje
 knjige i razdvoje knjige. @mailovi mas, knjige
 na je razdvoje u knjige korjenicko masi.

(b)

1. Nov sistem koji se koristi za razdvoje na masi
 razdvoje.
 a) Razdvoje knjige, kao je sistem koji se koristi
 razdvoje knjige, kao je sistem koji se koristi
 koji je razdvoje knjige u knjige.
 b) Razdvoje knjige, kao je sistem koji se koristi
 razdvoje knjige, kao je sistem koji se koristi
 koji je razdvoje knjige u knjige.
 c) Razdvoje knjige, kao je sistem koji se koristi
 razdvoje knjige, kao je sistem koji se koristi
 koji je razdvoje knjige u knjige.

(c)

6. da in delu razdvoje
 masi i knjige
 7. knjige razdvoje na
 razdvoje
 koje se knjige (razdvoje knjige)
 1. knjige razdvoje na knjige
 2. knjige
 3. knjige
 4. knjige na knjige
 5. knjige u knjige

(d)

Fig.5. Excerpt from multi-line handwritten text dataset: (a) Serbian Latin script, (b) Serbian Cyrillic script, (c) English script, and (d) Combination of Serbian and English Latin script.

Initial connected components CC_{init} represent the starting number of connected components in a sample text before applying the algorithm. However, after its application to the sample text, the number of connected components is counted. This number represents detected connected components CC_{det} . Further, the desired number of connected components CC_{ref} represents the number of text lines in the sample text. It is called referent number of connected components. By comparing the referent and detected number of connected components per each line of text, the algorithm efficiency is evaluated.

3.1. CLASSIFICATION OF THE TEXT SEGMENTATION ERRORS

If the number of obtained connected components in distinct text line is equal to one, then $CC_{det} = CC_{ref}$, which leads to a correctly segmented text line. The number of correctly detected text lines in the sample text is marked as $CC_{corrindet}$. However, all others are defined as error. Segmentation errors are present in the following circumstances:

- Over-segmentation detected text lines $CC_{overlindet}$,
- Under-segmentation detected text lines $CC_{underlindet}$.

Over-segmentation is a phenomenon that defines so-called split lines error. It represents the text lines, which are divided wrongly by the algorithm in two or more connected components [11]. In contrast, under-segmentation represents joined lines error. It corresponds to the situation where the sequence of n consecutive text lines is considered by the algorithm as a unique line [11]. These errors are illustrated in Fig.6.

3.2. ALGORITHM'S EVALUATION BASED ON BINARY CLASSIFICATION

Binary classification is based on the signal detection theory postulate [18]. Its task is to classify the members of a given set of connected components into two groups. The classifying is based on whether they have some property or not. It is represented by confusion matrix which is shown in Table 1.

Table 1. Confusion matrix.

Reality on Signal	Evaluation of the Signal	
	Yes	No
Present	Hit (H or TP)	Miss (M or FP)
Absent	False Alarm (FA or FN)	Correct Rejection (CR or TN)

Hence, if some connected components have a property and the test confirms it, then those connected components will represent true positives (TP). If some connected components do not have a property, but the test confirms it, then they will represent false negatives (FN). However, if some connected components have property, but the test mistakenly does not confirm it, then they will represent false positives (FP). Finally, if some connected components do not have a property, and the test confirms it, then they will represent true negatives (TN). In the context of classification tasks, all previous statements are used to compare the given classification of an item. Correlation of the previous statements with the testing results of the algorithm is as follows [19]:

- TP represents segmented text line hits $CC_{corrindet}$,
- FP represents segmented text line misses, i.e., $1 - CC_{corrindet} = CC_{overlindet} + CC_{underlindet}$, and
- FN represents the number of the false segmented text lines, i.e., $CC_{underlindet}$.

From these elements, the common evaluation measures can be extracted [18-20]:

- precision,
- recall, and
- f-measure.

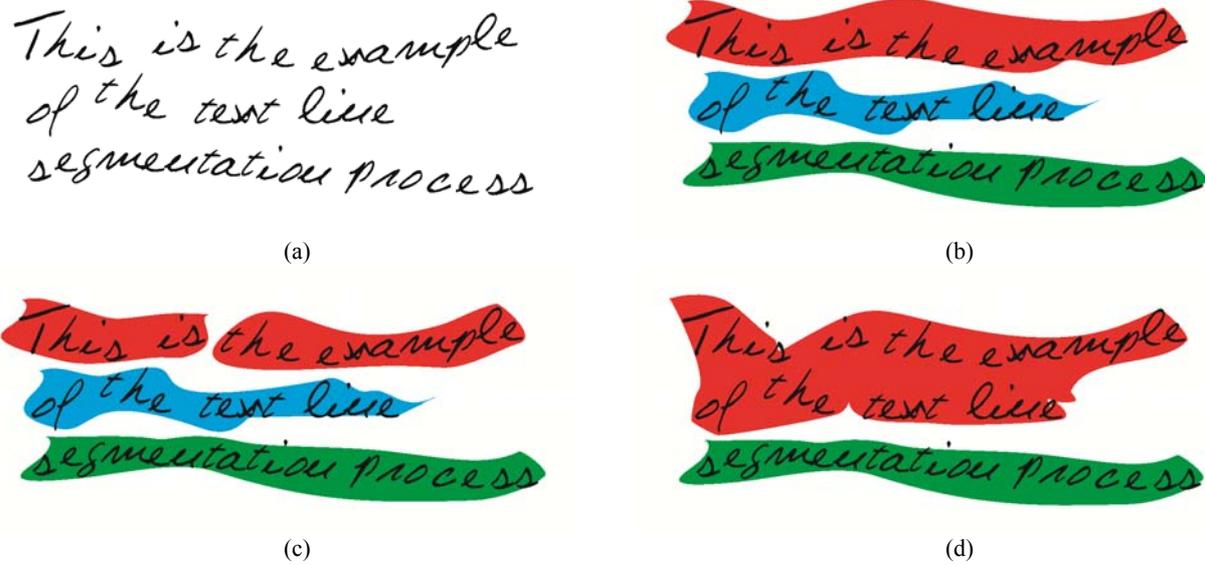


Fig.6. Text line segmentation: (a) Sample text, (b) Correctly segmented text lines, (c) Over-segmented text lines, and (d) Under-segmented text lines.

Precision is a measure of the ability of a system to present only relevant items. It is defined as [18-20]:

$$precision = \frac{TP}{TP + FP} \quad (1)$$

A higher *precision* means less false positives, while a lower *precision* means more false positives.

Recall is a measure of the ability of a system to present all relevant items. It is defined as [18-20]:

$$recall = \frac{TP}{TP + FN} \quad (2)$$

Higher *recall* means less false negatives, while lower *recall* means more false negatives.

Precision and *recall* can be combined to produce a single metric known as *f-measure*, which is the weighted harmonic mean of *precision* and *recall*. It is defined as [18-20]:

$$f\text{-measure} = 2 * \frac{precision * recall}{precision + recall} \quad (3)$$

These elements can be used as common evaluation measures.

Furthermore, as an extension to algorithm evaluation based on binary classification, the fourth measure is introduced. It is called $RMSE_{seg}$. It represents comparison of the number of detected and referent connected components per each text line. Hence, the number of referent connected components per line is equal to one. The variance evaluation is given $RMSE_{seg}$ [12-13]:

$$RMSE_{seg} = \sqrt{\frac{1}{N} \sum_{i=1}^N (CC_{i,ref} - CC_{i,det})^2} \quad (4)$$

where N is the total number of lines in the sample text, $CC_{i,ref}$ is the number of referent connected components in the text line i (equal to one per each line), and $CC_{i,det}$ is the number of detected connected components in the text line i . The importance of this additional measure is illustrated in the Appendix.

4. TESTING ALGORITHM

For testing purposes, smearing method [21] called algorithm based on Gaussian anisotropic kernel is used [5, 22-23]. This algorithm will be briefly explained. Its main task is expanding the 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 used 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. Thus, 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 the black pixel. Now the sentence begins with intensity that depends completely on the distance from the original black pixel. Newly formed pixels are grayscale. Hence, a 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 an ellipse in x and y direction. Consequently, σ represents standard deviation defining the curve spread parameter. Converting all these pixels into black pixels as well as inverting image, forms the new black pixel expanded areas [24]. These areas are named boundary-growing areas. A

similar method is given in [23]. In addition, parameter λ is introduced. It represents the ratio between L and K , i.e., $\lambda=L/K$. With the prior knowledge that a text line is a horizontally elongated shape [22], enhanced smearing effect in y -direction (horizontal axis) of the document image is suitable. Thus, for the algorithm based on Gaussian anisotropic kernel, $\lambda > 1$ is mandatory [23].

5. RESULTS, COMPARATIVE ANALYSIS, AND DECISION-MAKING

The main purpose of the testing is the process of the algorithm evaluation and parameter optimization. For the algorithm based on Gaussian anisotropic kernel, parameters of interest are those that define kernel size, i.e., K and λ . Text samples include letters with height up to 60 pixels. According to that, K should be chosen from 10% to 20% of the letter height [25]. Obtained results for different text samples are presented in Tables 2-4. However, only the best results are presented. Hence, only parameter pairs (K, λ) from the following set: (5, 5), (8, 3), (8, 4), (8, 5), (10, 3), (10, 4) and (10, 5) are given.

From Tables 2-4, the best results concerning *precision*, *recall* as well as *f-measure* are those obtained using parameter pair (K, λ) from the following set: (8, 4), (8, 5), (10, 4) and (10, 5). Obviously, using higher K (8 or 10) leads to better results for waved and fractured text. However, the results for straight text are quite disappointing. On the contrary, the choice of the parameter pair equal to (8, 5) brings uniform measure values. This is obtained by setting the criteria of the *f-measure* $> 70\%$ for all tests. In addition, the algorithm shows clear advantages for the text samples established on more complex text baseline, e.g., waved as well as fractured text due to *f-measure* equal to 98% and 88%, respectively.

At the end, $RMSE_{seg}$ (see Table 5) confirms results for the best choice of kernel size parameters. However, to improve the behavior of the testing algorithm, additional algorithm for the evaluation 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 *f-measure* and smaller $RMSE_{seg}$ values.

Finally, for the text samples that include letters with height up to 60 pixels, the optimal kernel size is $2K+1 \times 2(K*\lambda)+1$, i.e., $2 \times 8 + 1 \times 2 \times (8 \times 5) + 1$ px. This leads to $K \approx 15\%$ of letter height [25] and $L \approx 5 \times K$.

Table 2. Multi-line straight text segmentation test (K in pixels).

K	5	8			10		
λ	5	3	4	5	3	4	5
<i>precision</i> (%)	96.00	96.00	85.00	73.00	81.00	65.00	58.00
<i>recall</i> (%)	98.00	96.00	85.00	73.00	81.00	65.00	58.00
<i>f-measure</i> (%)	97.00	96.00	85.00	73.00	81.00	65.00	58.00

Table 3. Multi-line waved text segmentation test (K in pixels).

K	5	8			10		
λ	5	3	4	5	3	4	5
<i>precision</i> (%)	6.00	6.00	63.00	96.00	58.00	100.00	100.00
<i>recall</i> (%)	100.00	100.00	100.00	100.00	100.00	100.00	100.00
<i>f-measure</i> (%)	12.00	12.00	77.00	98.00	74.00	100.00	100.00

Table 4. Multi-line fractured text segmentation test (K in pixels).

K	5	8			10		
λ	5	3	4	5	3	4	5
<i>precision</i> (%)	2.00	6.00	75.00	88.00	56.00	83.00	81.00
<i>recall</i> (%)	33.00	60.00	90.00	88.00	84.00	83.00	81.00
<i>f-measure</i> (%)	4.00	11.00	82.00	88.00	68.00	83.00	81.00

Table 5. $RMSE_{seg}$ for different text samples (K in pixels).

K	5	8			10		
λ	5	3	4	5	3	4	5
<i>Straight text</i>	0.20	0.20	0.38	0.52	0.43	0.60	0.65
<i>Waved text</i>	2.46	2.61	0.66	0.20	0.85	0.00	0.00
<i>Segmentation text</i>	3.18	3.42	0.61	0.35	1.34	0.41	0.43

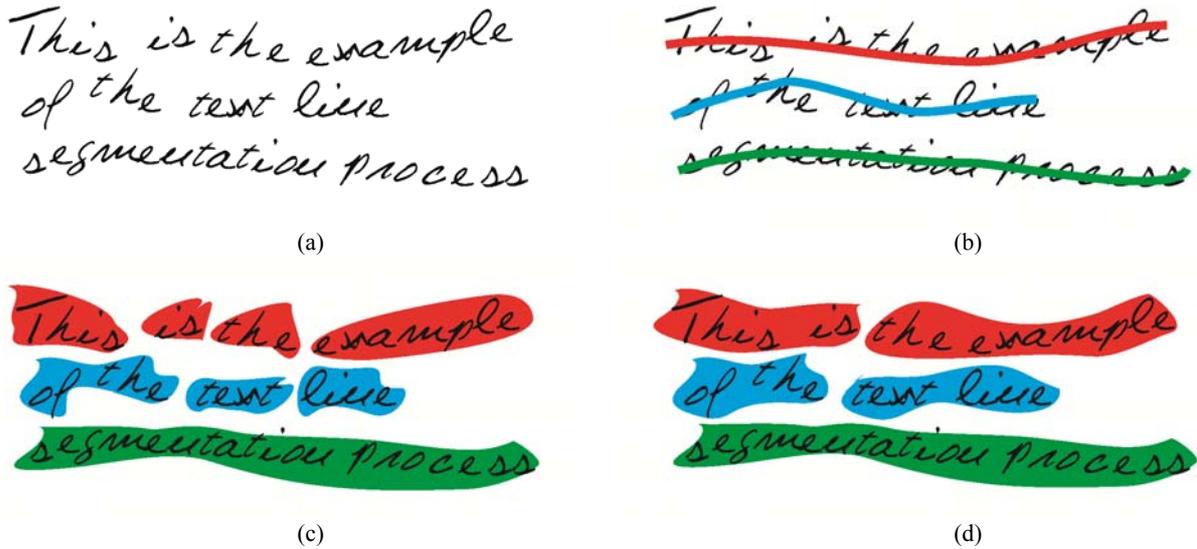


Fig.7. Text line segmentation: (a) Initial text, (b) Initial text with referent objects, (c) Text after segmentation made by the algorithm #1, and (d) Text after segmentation made by the algorithm #2.

6. CONCLUSIONS

This paper describes the methodology for the evaluation of the algorithms for text line segmentation. The proposed method is a step toward generalization of the test procedure in the domain of document image processing algorithm evaluation. It consists of the test experiments that incorporate various multi-line text experiments, i.e., straight, waved, fractured, and handwritten ones. Each of them consists of different script types. Algorithm under test is applied to each of the given text sets. Furthermore, the text line segmentation results are obtained. For its evaluation, a method established on extended binary classification is proposed. It is based on text segmentation errors classified by distinct measures such as *precision*, *recall*, and *f-measure*. In addition, it is extended with $RMSE_{seg}$ measure. However, the results from all test experiments are inter-related. Hence, they can be combined after some decision-making. The benefit of this process is the subset of the algorithm parameter's values, which are the starting point for the algorithm optimization. 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

To understand clearly the purpose of the $RMSE_{seg}$ measure, two different segmentation results are evaluated by it. After the procedure of text line segmentation by the algorithms #1 and #2, obtained results are shown in Fig.7.

All test results from algorithm #1 and #2 are reorganized according to segmentation binary classification. Such results are given in Table 6.

Table 6. Text line segmentation test results.

Algorithm	#1	#2
<i>precision</i> (%)	33.00	33.00
<i>recall</i> (%)	100.00	100.00
<i>f-measure</i> (%)	49.00	49.00
$RMSE_{seg}$	1.20	0.47

According to $RMSE_{seg}$, algorithm #2 shows slightly better performances than algorithm #1 in the domain of text line segmentation.

Further, various fragments of different sample text scripts from the test process are given in Fig.8-10.

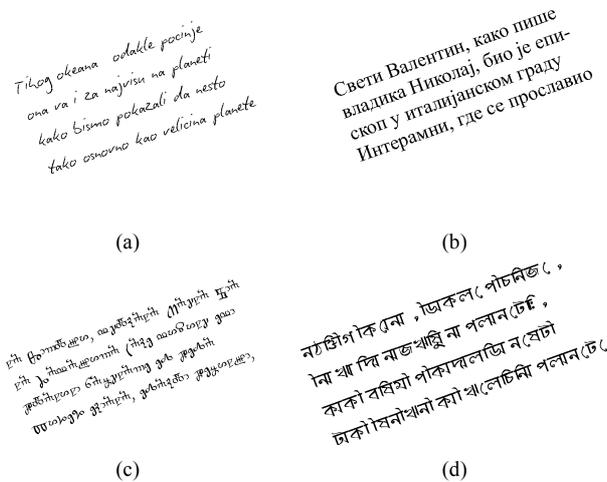


Fig.8. Multi-line straight text: (a) Latin text, (b) Serbian Cyrillic text, (c) Glagolitic text, and (d) Bengali text.

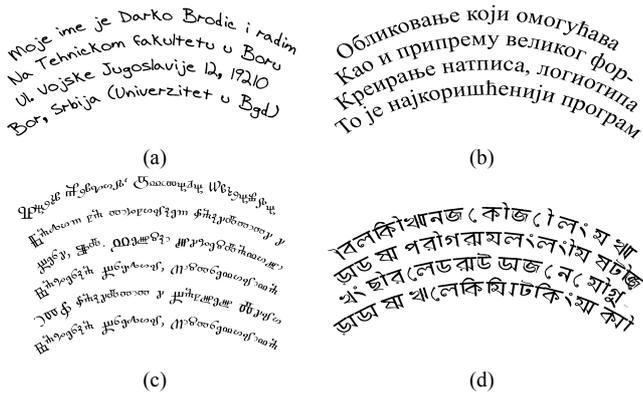


Fig.9. Multi-line waved text: (a) Latin text, (b) Serbian Cyrillic text, (c) Glagolitic text, and (d) Bengali text.

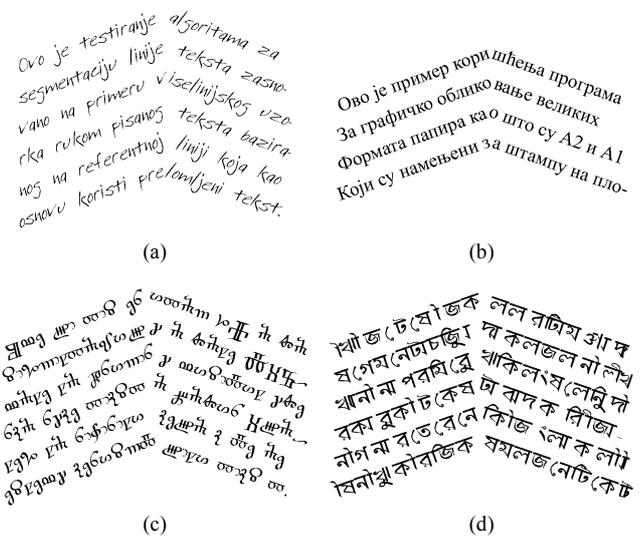


Fig.10. Multi-line fractured text: (a) Latin text, (b) Serbian Cyrillic text, (c) Glagolitic text, and (d) Bengali text.

ACKNOWLEDGMENT

This paper was supported by the Grant of the Ministry of Science and Technological Development of Republic of Serbia, as a part of the Project TR 33037 within the framework of Technological Development Program.

REFERENCES

[1] Du, X., Pan, W., Bui, T.D. (2009). Text line segmentation in handwritten documents using Mumford–Shah model. *Pattern Recognition*, 42 (12), 3136-3145.

[2] Likforman Sulem, L., Zahour, A., Taconet, B. (2007). Text line segmentation of historical documents: a survey. *International Journal on Document Analysis and Recognition*, 9 (2), 123-138.

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

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

[5] Yi, L., Yefeng, Z., Doermann, D., Jaeger, S. (2008). Script-Independent text line segmentation in freestyle handwritten documents. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30 (8), 1313-1329.

[6] Basu, S., Chaudhuri, C., Kundu, M., et. al. (2007). Text line extraction from multi-skewed handwritten document. *Pattern Recognition*, 40 (6), 1825-1839.

[7] Marti, U.V., Bunke, H. (2002). The IAM-database: an English sentence database for off-line handwriting recognition. *Journal on Document Analysis and Recognition*, 5 (1), 39-46.

[8] Sarkar, R., Das, N., Basu, S., Kundu, M., Nasipuri, M., Basu, D. (2011). CMATERdb1: a database of unconstrained handwritten Bangla and Bangla–English mixed script document image. *International Journal on Document Analysis and Recognition*, 14 (1), 25-33.

[9] Gatos, B., Stamatopoulos, N., Louloudis, G. (2011). ICDAR2009 handwriting segmentation contest. *International Journal on Document Analysis and Recognition*, 14 (1), 1-13.

[10] Louloudis, G., Gatos, B., Pratikakis, I., Halatsis, C. (2009) Text line and word segmentation of handwritten documents. *Pattern Recognition*, 42 (12), 3169-3183.

[11] Sanchez, A., Suarez, P.D., Mello, C.A.B., Oliveira, A.L.I., Alves, V.M.O. (2008). Text line segmentation in images of handwritten historical documents. In *Proceedings of the First Workshops on Image Processing Theory, Tools and Applications - IPTA 2008*. Sousse, Tunisia, 1-6.

[12] Brodić, D., Milivojević, D.R., Milivojević, Z. (2010). Basic test framework for the evaluation of text line segmentation and text parameter extraction. *Sensors*, 10 (5), 5263-5279.

[13] Brodić, D. (2010). Basic experiments set for the evaluation of the text line segmentation. *Przegład Elektrotechniczny*, 86 (11), 353-357.

[14] Brodić, D. (2011). Advantages of the extended water flow algorithm for handwritten text line segmentation. In Kuznetsov, S.O., et al. (eds.) *Pattern Recognition and Machine Intelligence*, LNCS, Vol. 6744. Berlin-Heidelberg: Springer, 418-423.

[15] Khashman, A., Sekeroglu, B. (2008). Document image binarisation using a supervised neural network. *International Journal of Neural Systems*, 18 (5), 405-418.

[16] Brodić, D. (2011). The evaluation of the initial skew rate for printed text. *Journal of Electrical Engineering - Elektrotechnický časopis*, 62 (3), 134-140.

[17] Zramdini, A., Inglod, R. (1993). Optical font recognition from projection profiles. *Electronic Publishing*, 6 (3), 249-260.

[18] Swets, J.A. (1988). Measuring the accuracy of diagnostic systems. *Science (New Series)*, 240 (4857), 1285-1293.

- [19] Qian, X., Liu, G., Wang, H., Su, R. (2007). Text detection, localization, and tracking in compressed video. *Signal Processing: Image Communication*, 22 (9), 752-768.
- [20] Bukhari, S.S., Shafait, F., Bruesl, T.M. (2009). Adaptive binarization of unconstrained hand-held camera-captured document images. *Journal of Universal Computer Science*, 15 (18), 3343–3363.
- [21] Shi, Z., Govindaraju, V. (2004). Line separation for complex document images using fuzzy runlength. In *Proceedings of the International Workshop on Document Image Analysis for Libraries*. Palo Alto, U.S.A.
- [22] Manmatha, R., Rothfeder, J.L. (2005). A scale space approach for automatically segmenting words from historical handwritten documents. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27 (8), 1212-1225.
- [23] Brodić, D., Milivojević, Z. (2010). Optimization of the Gaussian kernel extended by binary morphology for text line segmentation. *Radioengineering*, 19 (4), 718-724.
- [24] Razak, Z., Zulkiflee, K., et al. (2008). Off-Line handwriting text line segmentation: a review. *International Journal of Computer Science and Network Security*, 8, 12-20.
- [25] Brodić, D. (2010). Optimization of the anisotropic Gaussian kernel for text segmentation and parameter extraction. In Calude, C.S., Sassone, V. (eds.) *Theoretical Computer Science*, IFIP AICT, Vol. 323. Berlin-Heidelberg: Springer, 140-152.

Received May 24, 2011.
Accepted August 3, 2011.