2014 22nd International Conference on Pattern Recognition

Robust Text Line Segmentation for Historical Manuscript Images Using Color and Texture

Kai Chen*, Hao Wei*, Marcus Liwicki*†, Jean Hennebert*‡, and Rolf Ingold*
*DIVA (Document, Image and Voice Analysis) research group
Department of Informatics, University of Fribourg, Switzerland
Email: {firstname.lastname}@unifr.ch
†DFKI - German Research Center for Artificial Intelligence
‡University of Applied Sciences, HES-SO//FR, Bd. de Pérolles 80, 1705 Fribourg, Switzerland

***Abstract*—In this paper we present a novel text line segmentation method for historical manuscript images. We use a pyramidal approach where at the first level, pixels are classified into: *text*, *background*, *decoration*, and *out of page*; at the second level, text regions are split into *text line* and *non text line*. Color and texture features based on Local Binary Patterns and Gabor Dominant Orientation are used for classification. By applying a modified Fast Correlation-Based Filter feature selection algorithm, redundant and irrelevant features are removed. Finally, the text line segmentation results are refined by a smoothing post-processing procedure. Unlike other projection profile or connected components methods, the proposed algorithm does not use any script-specific knowledge and is applicable to color images. The proposed algorithm is evaluated on three historical manuscript image datasets of diverse nature and achieved an average precision of 91% and recall of 84%. Experiments also show that the proposed algorithm is robust with respect to changes of the writing style, page layout, and noise on the image.**

***Keywords*—Document Understanding, Segmentation, features and descriptors, Texture and color analysis,**

I. Introduction

Nowadays, a large number of historical documents have been digitized and made available to the public. With the increasing availability of computers and text-based software, the analysis of such documents is leveraged to higher dimensions leading to novel interests in digital humanities research. Historical Document Image Analysis and Recognition (HDIAR) methodologies are now widely used to enable computers to recognize the text content of documents. The vision of HDIAR is the automatic extraction of the information contained in a document; this includes the actual textual (and pictorial) information as well as writer identification, word spotting, or meta data.

A. State of the Art

In recent years, some HDIAR systems have been developed. MEMORIAL [2] aims at the development of a digital document workbench enabling the creation of distributed virtual archives based on printed historical document images. DEBORA [12] aims at improving the accessibility of rare sixteen century books through the Internet. It uses image analysis to extract documents meta-data. Efficient compression is realized by analysis of their content. AGORA [18] uses two maps to segment historical document images: a map of the foreground and a map of the background. After segmentation a list of blocks are created. Users are then able to label, merge and remove blocks. For non-latin HDIAR systems, Kitadai et al. [11] developed a system to support reading mokkans[1]. Functions as image processing, character recognition, context processing and communication between archaeologists are provided in the system.

Text line segmentation is performed in most of the systems mentioned above. This task is especially challenging in historical document image analysis, due to the low quality, degradation of the images, various distribution of elements such as holes, spots, decorations, complex layout, etc. The main methods of the text line segmentation are based on using projection profile, smearing algorithms, grouping connected components, and Hough transform [13]. In [14], text line segmentation is achieved by applying Hough transform on a subset of connected components of the document image. Some false alarms are corrected by a merging technique over the result of the Hough transform. The unclassified connected components are examined to determine if a new line is detected. Finally, the closest lines are grouped. In [20], the authors proposed a bottom-up method for text line segmentation in unconstrained handwritten Chinese documents. The authors group the connected components (CCs) of document images into a tree structure with a given distance metric. The distance metric is obtained by supervised learning on a dataset of pairs of CCs which makes the algorithm robust to handle documents with multi-skewed and curved text lines. Li et al. [10] convert a binary image into a probability map where each pixel represents the probability that it belongs to a text line. Based on the probability map, the boundaries of text lines are further determined by using region growth techniques. The algorithm has been applied on binary TIFF images.

B. Contribution

Most of the existing methods presume binary or gray scale images [4], [13]. In this paper, we propose a novel text line segmentation algorithm applicable to color historical manuscripts. In summary, the main novelties of our approach consist of (1) extension of the pyramidal approach presented in [3], (2) using color and texture information of pixel neighbors to train the classifier for predicting the class label, (3) feature selection in order to reduce the computational time and (4) post-processing procedure applied to refine the results. Experiments

[1]A mokkan is a wooden tablet with text written using a brush in Indian ink.

1051-4651/14 $31.00 © 2014 IEEE
DOI 10.1109/ICPR.2014.514

are achieved on three different historical manuscript image datasets [6]: *Parzival*[2], *George Washington*[3], and *Saint Gall*[4]. The results demonstrate that the proposed method is effective and robust to changes of writing style, page layout, and noise on the image. We conclude experimentally that color and texture are crucial information for text line segmentation on historical manuscript images.

The rest of the paper is organized as follows. Section II describes the proposed text line segmentation algorithm. Section III reports on the experimental results and Section IV presents conclusions and future works.

II. System Description

We consider text line segmentation as a pixel classification problem. Due to the large size of the images (at least 1500 $\times$ 2000 pixels for each image), layout analysis is time consuming. As our text line segmentation method will be embedded into a GUI, the algorithm will be used online and have to be computationally efficient. For this reason, we based our work on the pyramidal approach of [3]. At the first level, we scale each image to smaller size with scale factor $\alpha < 1.0$. As explained in [3], the scaled image is segmented into four parts, i.e., *out of page*, *background*, *text block*, and *decoration*. We did not extend and modify this level in our work. At the second level, the image has the double resolution of the first level in order to perform the more precise task of text line segmentation. Our contribution is mainly focused on this second level.

The proposed method for text line segmentation is based on machine learning requiring a training and a testing step. The advantage is that each procedure is modular allowing for the test of different features and classifiers. Figure 1 gives the overview of our system.

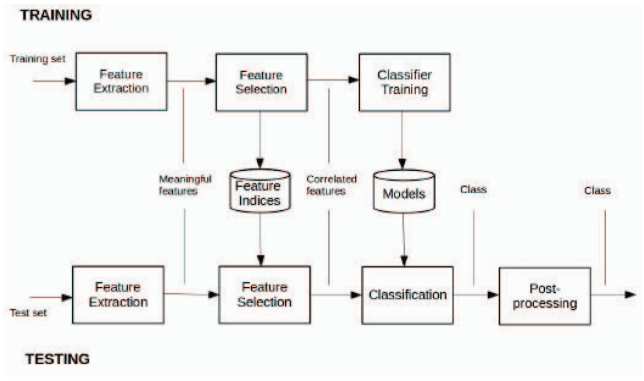


Fig. 1: Machine learning based text line segmentation approach diagram.

A. Feature Extraction

Feature extraction is an important part for the classification task. In the proposed method, each pixel $p_{x,y}$ is represented by a d-dimensional real-valued feature vector which is computed from its neighbors. For a given pixel $p_{x,y}$, its neighbors $NE_{p_{x,y}}$ are the pixels in a $n \times n$ window, $NE_{p_{x,y}}$ is defined as: $NE_{p_{x,y}} = \{p_{x',y'} | x' = x-d, x-d+1, \cdots, x+d-1, x+d \wedge y' = y-d, y-d+1, \cdots, y+d-1, y+d \wedge x'! = x \wedge y'! = y \wedge d = (n-1)/2\}$. Figure 2 depicts our feature extraction strategy. The following features are used.

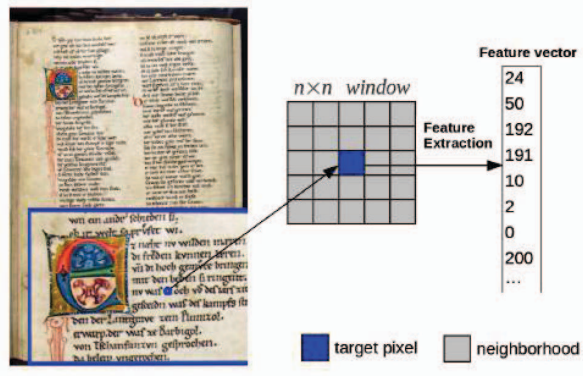


Fig. 2: Feature extraction diagram.

Color and coordinate features were successfully applied in our previous work [19]. In order to improve the performance, we have added several new features using the color information in our feature set. The feature vector of a pixel $p_{x,y}$ is composed as follows:

1) Coordinates x and y.
2) Values of primary colors of r, g and b.
3) Sum of neighborhood primary color values of r, g, and b. e.g., the sum of r component is given as $S(p_{x,y})^r = \sum_{i=-d}^{d} \sum_{j=-d}^{d} z^r_{x+i,y+j}$, where $d = (n-1)/2 = 2$ and $z^r_{x+i,y+j}$ is the r color component value at position $x+i, y+j$.
4) Horizontal and vertical maximum and minimum values of primary colors in its neighborhood.
5) Sum of neighborhood primary color values in the horizontal direction.
6) Sum of all pixels primary color values in the horizontal direction.
7) Mean value of neighborhood primary color, e.g., the mean value of r component is given as $M(p_{x,y})^r = \frac{1}{n \times n} S(p_{x,y})^r$.
8) Variance of neighborhood primary, e.g., the variance of r component is give as $V(p_{x,y})^r = \frac{1}{n \times n} \times \sum_{i=-d}^{d} \sum_{j=-d}^{d} (z^r_{x+i,y+j} - M(p_{x,y})^r)^2$.
9) Color smoothness [7] is a transformation of the variance. It is defined as $SMO(p_{x,y}) = \frac{1}{1+V(p_{x,y})}$.
10) Horizontal mean, variance, and smoothness of neighborhood primary color. We only use the neighbors on the horizontal position of $p_{x,y}$ to compute the values of the mean and the variance.
11) Laplacian [15] is the sum of second partial derivative of $z_{x,y}$ on x and y, where $z_{x,y}$ is the pixel value function on position x and y. It is defined as:

$$\nabla^2 z(x,y) = \frac{\partial^2}{\partial x^2} z(x,y) + \frac{\partial^2}{\partial y^2} z(x,y) \qquad (1)$$

Local binary patterns (LBP) has been successfully used for face recognition [1] and facial gesture recognition [22]. The histogram of the specific binary patterns is shown as

[2]http://www.parzival.unibe.ch
[3]http://memory.loc.gov/ammem/gwhtml/gwseries2.html
[4]http://www.e-codices.unifr.ch

an important feature-set [16] which can be used to represent the texture information of images. Based on this feature, we investigate using LBP for text line segmentation.

LBP is based on signs of differences of a circular neighboring pixels. It is defined as:

$$LBP_{P,R}(x,y) = \sum_{p=0}^{P-1} s(g_p - g_c)2^p$$
$$s(x) = \begin{cases} 1 & x \geq 1 \\ 0 & x < 0 \end{cases} \quad (2)$$

For a given pixel $p_{x,y}$, P is defined as the number of neighbors; R is defined as the value of radius on a circle; $p_{x,y}$ is the center point of that circle. g_c is the gray level value of the central pixel at coordinate x, y; g_p is the gray level value of the pth point on the circular neighborhood.

In order to capture the local structure information of each pixel and achieve rotation invariance, we compute the LBP histogram in its $n \times n$ neighbors based on the Rotation Invariant Unifrom Pattern (LBP^{riu2}) [8]. $LBP^{riu2}_{P,R}$ is defined as:

$$LBP^{riu2}_{P,R} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - gc) & \text{if } U(LBP_{P,R}) \leq 2 \\ P+1 & \text{otherwise} \end{cases} \quad (3)$$

where

$$U(LBP_{P,R}) = |s(g_{P-1} - g_c) - s(g_0 - g_c)| + \sum_{p=1}^{P-1} |s(g_p - g_c) - s(g_{p-1} - g_c)| \quad (4)$$

The feature vector of $p_{x,y}$ is defined as the histogram of the $LBP^{riu2}_{P,R}$ in its $n \times n$ neighborhood. The number of bins of the histogram is $P+2$. In our experiments, we use the combination of the LBP histograms in three different number of neighbors and radius: ${LBP_{(8,1)}}^{riu2}$, ${LBP_{(16,2)}}^{riu2}$, and ${LBP_{(24,3)}}^{riu2}$. The total number of bins of the histogram is 54.

Gabor dominant orientation. Gabor filters have been widely applied in classification tasks, e.g., texture analysis, face recognition, moving object tracking. It has also been used in the filed of character recognition [9]. Gabor filter performs a spatial frequency analysis. It extracts orientation-dependent frequency content. Based on this feature, we use the Gabor filter to detect the dominant orientation of each pixel. As described in [9], Gabor filters localize direction spatial frequency at orientation θ, i.e., the output of Gabor filter $h(x,y)$ to an image responds maximally at those edges of the orientation θ. For a given pixel $p_{x,y}$, in order to get the dominant orientation on its $n \times n$ neigborhood, we compute the sum of the convolution of a set of Gabor filters $h(x,y)$ on different orientations. We define $I(x,y,\theta)$ as the sum of convolution of Gabor filter $h(x,y)$ on an image $u(x,y)$ at the orientation θ, where $I(x,y,\theta)$ is defined as:

$$I(x,y,\theta) = \sum_{i=x-\frac{n-1}{2}}^{x+\frac{n-1}{2}} \sum_{j=y-\frac{n-1}{2}}^{y+\frac{n-1}{2}} u(i,j) e^{\frac{-(i-x)^2-(j-y)^2}{2\sigma^2}} e^{j\lambda(cos\theta(i-x)+sin\theta(j-y))} \quad (5)$$

The orientation angles of this set of Gabor filters are: $\theta_k | \theta_k = k \times \frac{2\pi}{n-1}$, where $k = 0, \cdots, (n-1)$. If we obtain a maximum output $I(x,y,\theta_k)$, then θ_k is considered as the dominant orientation angle of the pixel at position (x,y).

B. Feature Selection

Due to the large amount of pixels, a feature selection method is applied to reduce the computational time. The objective of the feature selection is to reduce the dimensionality of the feature space without decreasing the accuracy by removing irrelevant and redundant features. In our system, we use the Fast Correlation-Based Filter (FCBF) algorithm [21] which not only selects the features relevant to the classes but also removes the redundant features. FCBF is a filter model feature selection method. It generates a subset of features S', such that all the features in S' are highly correlated to the classes but they are uncorrelated to any of the other features. The correlation measure is based on the information-theoretical concept of *entropy*, a measure of the uncertainty of a random variable. In particular, a threshold δ is chosen to measure the *predominant correlation* for a feature F_i to a class C. For the definition of *predominant correlation* and details of FCBF feature selection algorithm, we refer to [21].

How to choose δ is crucial for this algorithm. If δ is too large some important features may be removed. On the other hand, if δ is too small, some redundant features may not be removed. The optimal δ value can be discovered by exhaustive experimental search. We propose here an alternative procedure to efficiently compute δ. We set δ to a very small value at the beginning, then we call the FCBF algorithm several times until the optimized feature set size become stable. Algorithm 1 gives the modified FCBF algorithm.

Algorithm 1 Modified FCBF algorithm

1: **procedure** MODIFIEDFCBF($S, c = 1, \delta = 0.0001, \gamma = 0.001$) ▷ S: the feature set. c: the threshold for testing if the optimized feature size is stable. δ: a threshold is chosen to measure the predominant correlation for a feature. γ: in each loop, we increase the value of δ, such that $\delta = \delta + \gamma$
2: $S_{best} \leftarrow FCBF(S, \delta)$ ▷ Get optimized feature set
3: **repeat**
4: $previousSize \leftarrow sizeof(S_{best})$
5: $\delta \leftarrow \delta + \gamma$
6: $S_{best} \leftarrow FCBF(S, \delta)$
7: $currentSize \leftarrow S_{best}$
8: $diffSize \leftarrow |currentSize - previousSize|$
9: **until** $diffSize \leq c$
10: **return** S_{best} ▷ The optimized feature set
11: **end procedure**

C. Classification

The focus of our work is in the proposition of efficient features and their selection as described above. Therefore, we used for the modelling part default settings of a state-of-the-art SVM implementation [5].

D. Post-Processing

We observe that the pixels between characters in a text line are usually degraded. By observing some text line segmentation results with the proposed method on some randomly selected images in the training set, we find some errors of classification consistently appears on these pixels. In order to correct this type of errors, we proposed a post-processing method as follows:

After text line classification, pixels are labeled into two classes: *text line* and *non text line*. For a *non text line* pixel p, we compute the number of *text line* pixels $NT(p_{x,y})$ in its $n' \times n'$ neighborhood. If $NT(p_{x,y}) \geq T$, then p is labeled to *text line*, where T is a pre-defined threshold value. The effectiveness of this method is demonstrated in Section III.

III. Experiments

The proposed system is evaluated on three different historical document image datasets.

1) *George Washington* data set consists of 20 pages written in English. These pages were taken from letters written by George Washington and his associates in the 18^{th} century.
2) *Saint Gall* data set consists of 60 manuscript pages from a medieval manuscript written in Latin. It contains the hagiography Vita sancti Galli by Walafrid Strabo.
3) *Parzival* data set [6] consists of 47 pages written by three writers. These pages were taken from a medieval German manuscript from the 13th century that contains the epic poem Parzival by Wolfram von Eschenbach.

The datasets are of very different nature. *George Washington* dataset consists of images of manuscript written with ink on paper and the images are in gray levels. The last two datasets consist of images of manuscripts written with ink on parchment and the images are in color, while the former suffers from many degradations. Figure 3 gives some example pages from the datasets.

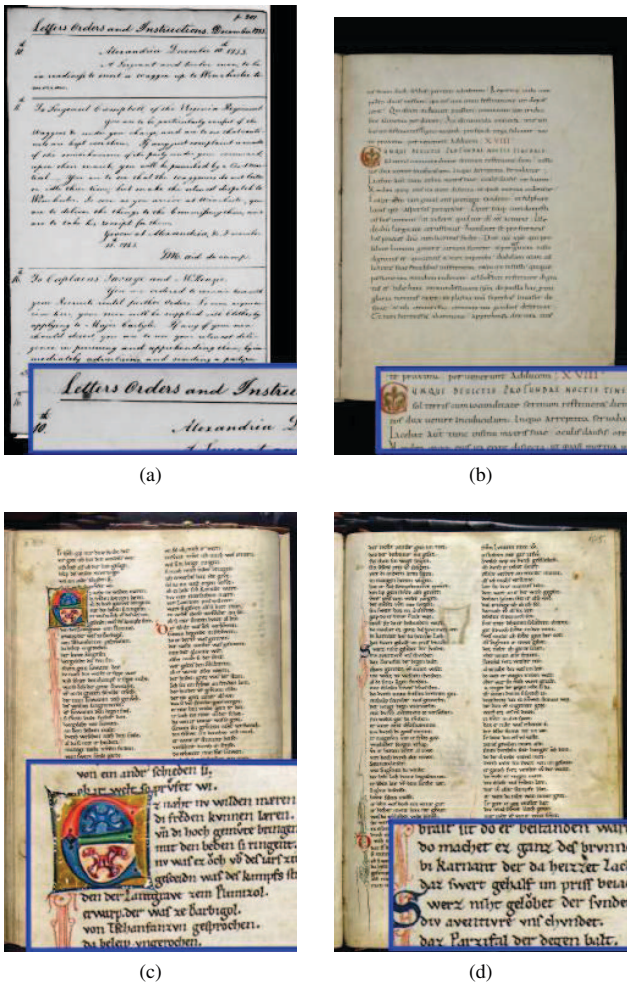
Table I gives the details of the training set TR and test set TE for each of the datasets. The scale factor is set to the value $\alpha = \frac{1}{16}$.

TABLE I: Training and test sets details.

	# pixels of original image	# pixels of scaled image	$\|TR\|$	$\|TE\|$
George Washington	2200×3400	275×425	10	5
Saint Gall	1664×2496	208×312	20	30
Parzival	2000×3008	250×376	24	14

A. Evaluation

As described in Section II, for each pixel, we choose its 11×11 neighbors to extract the features. The feature vector



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

Fig. 3: Page 3a and 3b from the *George Washington* and *Saint Gall* datasets respectively. Pages 3c and 3d from the *Parzival* dataset.

is composed of all the features described in Section II-A. For each dataset, we randomly select $20'000$ pixels in TR to create a validation set VS. In order to reduce the feature vector size, we apply the modified FCBF algorithm as described in Section II-B on VS to generate a optimal feature subset S'. In the post-processing phase, we set the threshold $T = 0.6$ and window size $n' = 3$. For each dataset, the proposed method is evaluated on four setups: (1) use all the features without post-processing. (2) use all the features with post-processing. (3) use selected features without processing. (4) use selected features with post-processing. For reporting the results we use the Precision P and Recall R values on the pixel level.

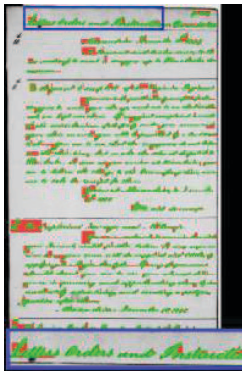
Figure 4 gives some examples of the segmentation results. The performance is given in Table II. We observe that the feature selection algorithm does not degrade the performance on the *Saint Gall* dataset and even improves the performance a little for the *George Washington* and *Parzival* datasets. Noteworthily, the dimension of feature vector is reduced on all datasets.

B. Analysis

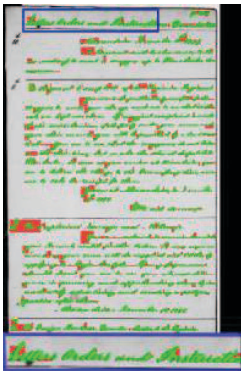
The experiments show the effectiveness and robustness of the proposed text line segmentation method. However, some

TABLE II: Text line segmentation results

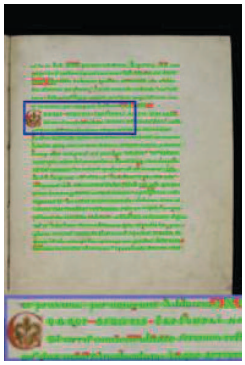
	George Washington			Saint Gall			Parzival		
	Feature size	P (%)	R (%)	Feature size	P (%)	R (%)	Feature size	P (%)	R (%)
All features (without post-processing)	111	87	69	111	84	89	111	89	88
All features (with post-processing)	111	92	72	111	86	91	111	92	90
Selected features (without post-processing)	41	88	67	50	84	88	43	89	87
Selected features (with post-processing)	41	94	71	50	86	91	43	93	91



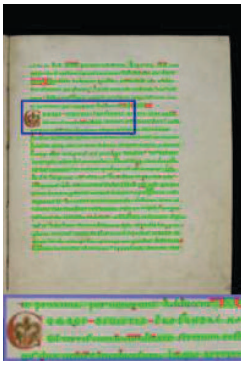
(a) $P = 87\%$, $R = 67\%$



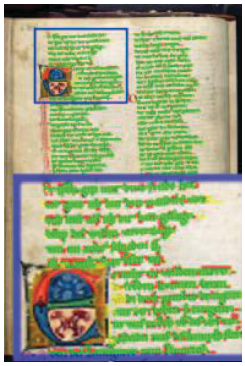
(b) $P = 94\%$, $R = 72\%$



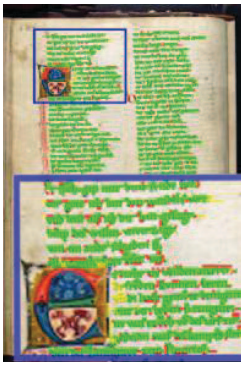
(c) $P = 84\%$, $R = 91\%$



(d) $P = 86\%$, $R = 93\%$



(e) $P = 91\%$, $R = 83\%$



(f) $P = 97\%$, $R = 89\%$

Fig. 4: 4a, 4c, and 4e are the text line segmentation results by using the proposed method on the selected feature set without post-processing. 4b, 4d, and 4f are the text line segmentation results with the post-processing algorithm. Green is used for correct prediction. Red is used for wrong prediction. Yellow is used for missing prediction.

types of errors appear consistently over the three datasets. We observe that due to the noise and writing style, the pixels on the contour and between strokes are difficult to be correctly classified. By using the post-processing algorithm described in Section II-D, the accuracy has been improved. Figure 5 shows some examples.

(a) (b) (c)

Fig. 5: Post-processing results. Green is used for the corrected pixels. 28.1%, 12.6%, and 18.0% misclassified pixels have been corrected in 5a, 5b and 5c respectively.

However there are still errors appearing on the contour of text lines. These errors are mainly caused by the noise on the border of the text line. To solve the problem, a pre-processing procedure could be applied to enhance the image by removing the noise. A post-processing procedure could also be applied to validate the pixels on the border. Figure 6 gives some examples of such errors.

IV. Conclusion

We proposed a novel method for text line segmentation on historical manuscript images. Our SVM-based system classifies each pixel into two classes: *text line* and *non text line*. The feature size is reduced by applying the modified FCBF algorithm. A post-processing procedure is used to refine the results. We evaluate this algorithm on three different datasets. Experiments demonstrate the effectiveness and robustness of the proposed method.

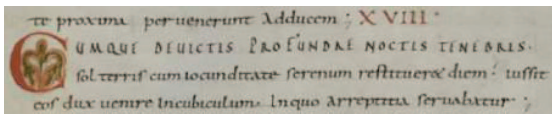
Our future work includes investigating the selection of more discriminative features, refining the post-processing procedure by parameter tuning, and choosing optimal window size for feature selection. We also plan to integrate this algorithm into a new historical document ground-truthing framework DIVADIA. DIVADIA aims at producing and modifying the ground-truth of the historical document images by incremental learning. Finally, it is planned to evaluate this algorithm on historical manuscript images of various languages.

Acknowledgment

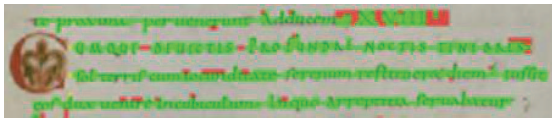
We would like to thank Michael Stolz for providing us with the Parzival data set. The project is part of HisDoc and HisDoc 2.0 funded by the SNF via the grant numbers CRSI22_125220 and 205120_150173.

(a)

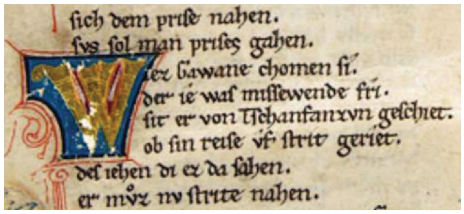
(b)



(c)



(d)



(e)

(f)

Fig. 6: Examples of errors at the border. 6b, 6d, and 6f give the text line segmentation results of 6a, 6c, and 6e.

REFERENCES

[1] T. Ahonen, A. Hadid, and M. Pietikäinen, *Face Description with Local Binary Patterns: Application to Face Recognition*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, issue. 12, 2006, pp. 2037-2041.

[2] A. Antonacopoulos and D. Karatzas, *Document Image Analysis for World War II Personal Records*, First International Workshop on Document Image Analysis for Libraries (DIAL), 2004, pp. 336-341.

[3] M. Baechler and R. Ingold, *Multi Resolution Layout Analysis of Medieval Manuscripts Using Dynamic MLP*, Proceedings of The Eleventh International Conference on Document Analysis and Recognition, 2011, pp. 1185-1189.

[4] I. Bar-Yosef, N. Hagbi, K. Kedem, and I. Dinstein, *Line Segmentation for Degraded Handwritten Historical Documents*, Internal Conference on Document Analysis and Recognition, 2009, pp. 1161-1165.

[5] C.-C. Chang and C.-J. Lin, *LIBSVM : a library for support vector machines*, ACM Transactions on Intelligent Systems and Technology, 2011, vol. 2, issue 3, pp. 2:27:1–27:27.

[6] A. Fischer, A. Keller, V. Frinken, and H. Bunke, *Lexicon-Free Handwritten Word Spotting Using Character HMMs*, Pattern Recognition Letters, vol. 33(7), 2012, pp. 934-942.

[7] M. Hangarge, K.C. Santosh, S. Doddamani, and R. Pardeshi, *Statistical Texture Features based Handwritten and Printed Text Classification in South Indian Documents*, CoRR, abs/1303.3087, 2013.

[8] Y. He, N. Sang, and R. Huang, *Local Binary Pattern histogram based Texton learning for texture classification*, Image Processing (ICIP), 2011, 18th IEEE International Conference, pp. 145-153.

[9] P. Hu, Y. Zhao, Z. Yang, and J. Wang, *Recognition of gray character using gabor filters*, Proceedings of the Fifth International Conference on Information Fusion, vol. 1, 2002, pp. 419-424.

[10] Y. Li, Y. Zheng, D. Doermann, and S. Jaeger, *Script-Independent Text Line Segmentation in Freestyle Handwritten Documents*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 8, 2008, pp. 1313-1329.

[11] A. Kitadai, J. Ishikawa, M. Nakagawa, M. Baba, and H. Watanabe, *Document Image Retrieval to Support Reading Mokkans*, The Eighth IAPR International Workshop on Document Analysis Systems, 2008, pp. 533-538.

[12] F. Le Bourgeois and H. Emptoz, *DEBORA: Digital AccEss to BOoks of the RenAissance*, International Journal of Document Analysis and Recognition (IJDAR), vol. 9, issue 2-4, 2007, pp. 193-221.

[13] L. Likforman-Sulem, A. Zahour, and B. Taconet, *Text line segmentation of historical documents: a survey*, International Journal of Document Analysis and Recognition (IJDAR), vol. 9, issue 2-4, 2007, pp. 123-138.

[14] G. Louloudis, B. Gatos, I. Pratikakis, and C. Halatsis, *Text Line and Word Segmentation of Handwritten Documents*, Pattern Recognition, vol. 42, issue 12, 2009, pp. 3169-3183.

[15] M. Seuret, *Printed Content & Handwritten Notes Discrimination in Scanned Documents*, Master Project, University of Fribourg, Department of Informatics, 2013.

[16] T. Ojala, M. Pietikäinen, and T. Maenpaa, *A Comparative Study of Texture Measures with Classification Based on Feature Distributions*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, issue. 7, 2002, pp. 971-987.

[17] J.R. Quinlan, *Induction of decision trees*, Machine Learning, vol. 1, issue 1, 1986, pp. 81-106.

[18] J.Y. Ramel, S. Busson, and M. L. Demonet, *AGORA: the Interactive Document Image Analysis Tool of the BVH Project*, Proceedings of the Second International Conference on Document Image Analysis for Libraries (DIAL), 2006, pp. 145-155.

[19] H. Wei, M. Baechler, F. Slimane, and R. Ingold, *Evaluation of SVM, MLP and GMM Classifiers for Layout Analysis of Historical Documents*, International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 1220-1224.

[20] F. Yin and C. L. Liu, *Handwritten Chinese text line segmentation by clustering with distance metric learning*, Pattern Recognition, vol. 42, issue 12, 2009, pp. 31463157.

[21] L. Yu and H. Liu, *Feature selection for high-dimensional data: A fast correlation-based filter solution*, Proceedings of the Twentieth International Conference on Machine Learning, AAAI Press, 2003, pp. 856-863.

[22] G. Zhao and M. Pietikäinen, *Dynamic Texture Recognition Using Local Binary Patterns with an Application to Facial Expressions*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, issue. 6, 2007, pp. 915-928.