Border Noise Removal of Camera-Captured Document Images using Page Frame Detection

Syed Saqib Bukhari*, Faisal Shafait† and Thomas M. Breuel*
*Image Understanding and Pattern Recognition (IUPR)
Technical University of Kaiserslautern, Germany
†Multimedia Analysis and Data Mining (MADM)
German Research Center for Artificial Intelligence (DFKI), Kaiserslautern, Germany
bukhari@informatik.uni-kl.de, faisal.shafait@dfki.de, tmb@informatik.uni-kl.de

***Abstract*—Camera-captured document images usually contain two main types of marginal noise: textual noise (coming from neighboring pages) and non-textual noise (resulting from the page surrounding and/or binarization process). These types of marginal noise degrade the performance of the preprocessing (dewarping) of camera-captured document images and subsequent document digitization/recognition processes. Page frame detection is one of the newly investigated areas in document image processing, which is used to remove border noise and to identify the actual content area of document images. In this paper, we present a new technique for page frame detection of camera-captured document images. We use text and non-text contents information to find the page frame of document images. We evaluate our algorithm on the DFKI-I (CBDAR 2007 Dewarping Contest) dataset. Experimental results show the effectiveness of our method in comparison to other state-of-the-art page frame detection approaches.**

***Keywords*-Border Noise Removal, Page Frame Detection, Camera-Captured Document Images**

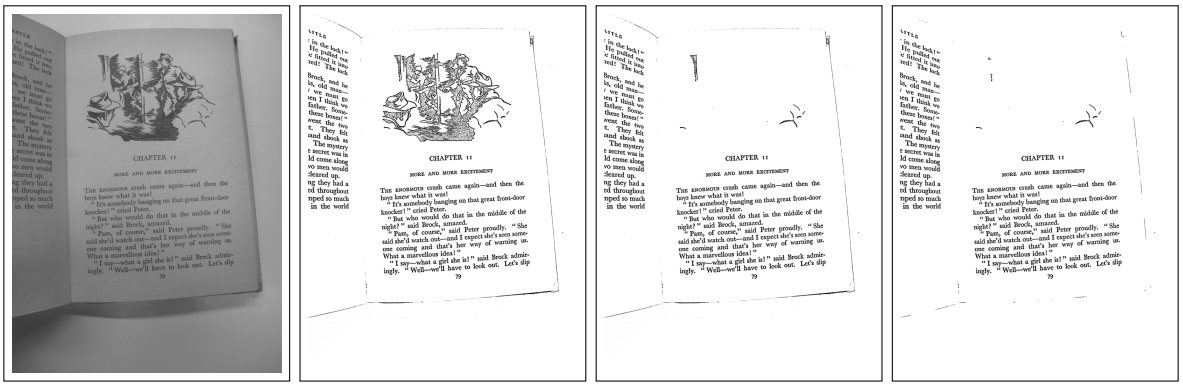
I. Introduction

When a page of a book is photographed, the captured image usually contains undesired parts of text from the neighboring page. Besides, some regions of background (table surface etc.) also appear in the image. These undesired regions of the image are usually referred to as border noise [1]. These types of border noise are called textual noise and non-textual noise, respectively. When textual noise regions are fed to a character recognition engine, extra characters appear in the output of the OCR system along with the actual contents of the document. These extra characters in the OCR output result in inaccurate retrieval results, since the keywords given by the user might match some text from the textual noise instead of the actual document contents. Non-textual noise, on the other hand, makes further processing of document like text-line extraction or dewarping a difficult task.

The problem of border noise is also well-known in the domain of scanned document analysis. Many approaches have been reported in literature to deal with border noise of scanned images. Most of the these approaches (e.g. [2], [3], [4]) focus only on removal of non-textual noise. Cinque et al. [5] propose an algorithm for removing both textual and non-textual noise from grayscale images based on image statistics like horizontal/vertical difference vectors and row luminosities. The method presented in [6] detects border noise using different black/white filters. These approaches rely on certain assumption about scanned documents (like an axis-aligned pattern of noise or presence of thick black non-textual noise regions). However, these assumptions do not hold for camera-captured documents since the document can be captured from any perspective (hence page border is not axis-aligned any more). Besides, the captured document is binarized using a local thresholding method like [7] (hence no thick black regions appear in the binarized image).

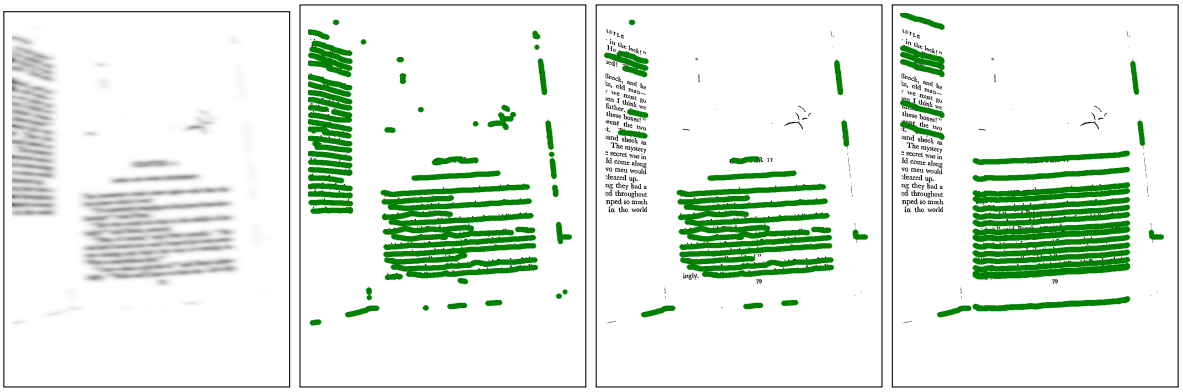
Instead of identifying and removing noisy components themselves, some methods focus on identifying the actual content area or the page frame of the documents [8], [9]. The page frame of a scanned document is defined as the smallest region (rectangle or polygonal) that encloses all the foreground elements of the document image. The method presented in [8] finds the page frame of structured documents (journal articles, books, magazines) by exploiting their text alignment property. The method by Fan et. al [10] estimates page frame using a rectangular active contour. This method is not directly applicable to page frame detection of camera-captured documents due to the presence of perspective distortions. Stamatopoulos et al. [9] proposed a method for splitting double-page scanned document images into two pages without noisy borders. Their method is based on vertical and horizontal white runs projections.

So far very few approaches are developed for camera-captured document images. Shafait et al. [8] applied their page frame detection approach to camera-captured document images. When applied to camera-captured document images, the method focuses on finding the left and right page border lines only using a geometric matching method. The method gives good results for camera-captured document images, but does not remove border noise on the upper and lower sides of the document images. Stamatopoulos et al. [11] proposed an algorithm for detecting borders of camera-captured document images based on projection profile. This



(a) grayscale image (b) binarized image (c) segmented text parts (d) after noise cleanup

Figure 1. Preprocessing: (a) a sample grayscale camera-captured document image, (b) binarized document image, (c) segmented text parts of the binarized image, (d) cleaned image after noise removal.



(a) smoothed image (b) detected ridges (c) cleaned/filtered ridges (d) aligned ridges

Figure 2. Text-Line Detection: (a) the smoothed image is generated using Gaussian filter bank smoothing, (b) ridges are detected from the smoothed image; most of the ridges represent text-lines, (c) small ridges and ridges near corners are removed using heuristically applied rules, (d) ridges have been aligned (with respect to their staring and ending positions) by projecting neighboring ridges over each of them.

method works well for a small degree of skew/curl in document images, but can not handle document images with a large degree of skew/curl, which is usually present in hand-held camera-captured documents.

In this paper we present a page frame detection method for camera-captured document images. The method starts with preprocessing which includes binarization and text and non-text segmentation steps. Then, text-lines are detected by applying the ridge based text-line finding method [12]. Finally, page frame is detected by using text-lines and text and non-text information. Our method can detect the upper and lower borders together with the left and right borders, and is robust to a large degree of skew/curl in camera-captured document images.

The rest of the paper is organized as follows. The proposed page frame detection method is described in Section II. Experiments and results are discussed in Section III. Section IV presents our conclusions.

II. Page Frame Detection Method

The proposed page frame detection method consists of three main steps: i) preprocessing, ii) text-line detection, iii) page frame detection. Preprocessing (binarization and text and non-text segmentation) of camera-captured document images is discussed in Section II-A. Text-line detection method is described in Section II-B. Page frame detection method using text-line and text and non-text contents information is explained in Section II-C.

A. Preprocessing

Our preprocessing approach mainly consists of binarization and text segmentation steps. An input grayscale camera-captured document image is first binarized using the adaptive thresholding technique mentioned in [13], which is described as follows: "for each pixel, the background intensity $B(p)$ is defined as the 0.8-quantile in a window shaped surrounding; the pixel is then classified as background if its intensity is above a constant fraction of $B(p)$". An example grayscale

document and its correspoding binarized document images are shown in Figure 1(a) and Figure 1(b), respectively.

We presented a multiresolution morphology based text and non-text segmentation algorithm in [14], that can segment text from different types of non-text elements like halftones, drawing, graphics, etc. In this approach, the resolution of an input (binary) document image is reduced iteratively by applying *threshold reduction* strategy for removing text elements and keeping non-text elements. The reduced image, after appropriate expansion, is used as non-text mask image. The segmented text form the binarized document image (Figure 1(b)) is shown in Figure 1(c).

After text and non-text segmentation, a heuristic size based noise cleanup process is applied for removing comparatively large (marginal noise) and small (salt-and-pepper noise) elements as follows. A connected component is considered as a large noisy component if its height/width is greater than 10% of document height/width or greater than 7 standard deviation above mean height/width. Similarly, a connected component is removed as a small noisy component if its area is smaller than $\frac{1}{3}^{rd}$ of the mean area. The document image in Figure 1(c) after noise cleanup is shown in Figure 1(d).

B. Text-Line Detection

We introduced a ridge based text-line extraction method for warped camera-captured document images in [12]. Our ridge based text-line finding method consists of two standard and easy to understand image processing algorithms: (i) Gaussian filter bank smoothing and (ii) ridge detection. The ridge based text-line detection method is briefly described here for the completeness of this paper.

First, the ranges for Gaussian filter's parameters, i.e. σ_x, σ_y and θ, are defined empirically for generating a set of filters. Then, the set of filters is applied to each pixel and the maximum output response is selected for the smoothed image. Figure 2(a) shows the smoothed version of the document image as shown in Figure 1(d). After smoothing, text-lines are extracted by detecting ridges from the smoothed image.

Most of the detected ridges, that are shown in Figure 2(c), are situated over text-lines. Some of them are also very small in size as compared to others, and some on them lie over marginal textual noise. A ridge is considered as a small-size ridge if its length is smaller than $\frac{1}{10}^{th}$ of document width. Textual-noise is usually present in the left and right corners of the document. Therefore, a ridge is considered as a ridge over textual-noise if its starting/ending point exists very close (within ± 25 pixels) to the left/right corner of document image and its length is smaller than $\frac{1}{5}^{th}$ of document width. After filtering small-size ridges and ridges over textual-noise, the starting and ending points of the remaining ridges can be used for approximating left and right borders, respectively. The remaining ridges are shown in Figure 2(c). Most of these remaining ridges are present over the actual content area of the page.

Another major problem in using these remaining ridges/text-lines for left and right borders approximation is that, their starting and ending positions are not aligned with respect to each other. We presented a ridges alignment method in [15] for solving this problem, which is described here as follows. For each ridge, the neighboring top and bottom ridges are projected over it, and then are combined together to produce a new (aligned) ridge. The aligned ridges are shown in Figure 2(d). Some more results of ridges after alignment for document images in DFKI-I dataset [16] are shown in Figure 3.

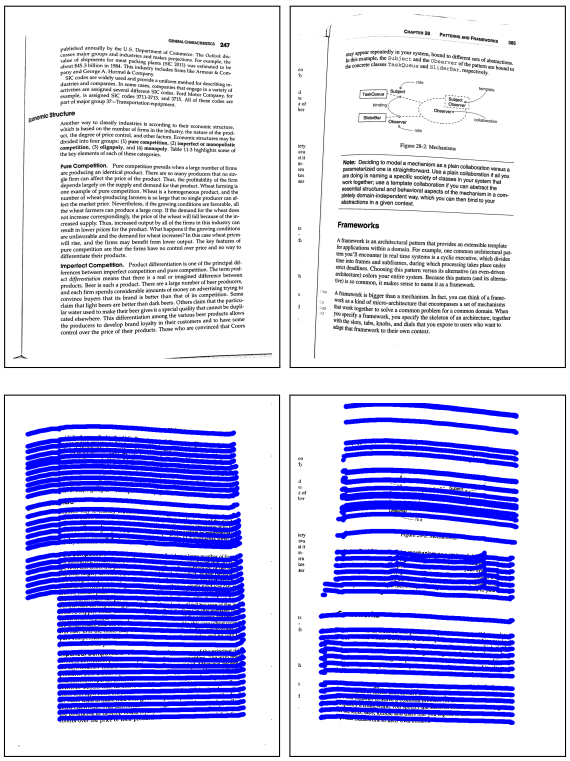
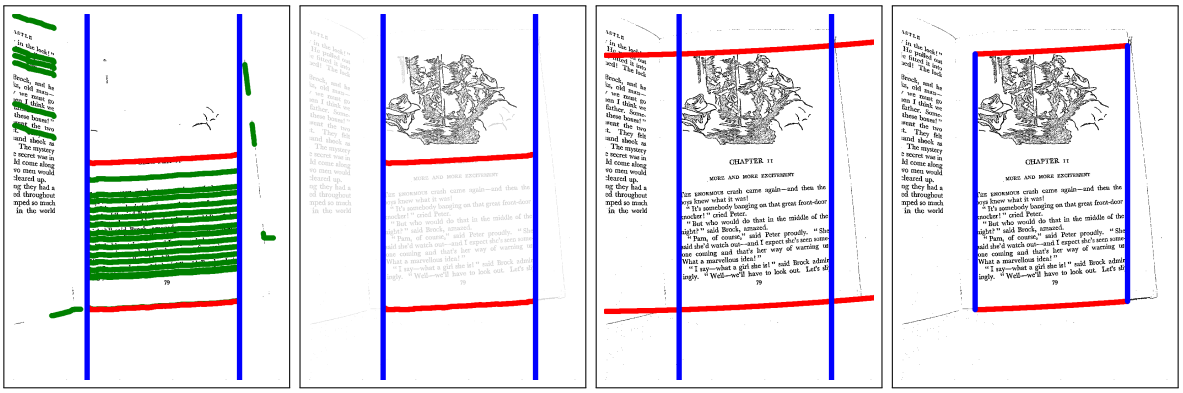


Figure 3. Sample results of aligned ridges for documents in DFKI-I dataset.

C. Page Frame Detection

The left and right borders are calculated by applying a straight-line approximation algorithm over the starting and ending points of the ridges, respectively. For this purpose, we have chosen RANdom SAmple Consensus (RANSAC) method, which approximates slope and intercept parameters. The left and right borders are shown in Figure 4(a) in blue color. The initial estimation of upper and lower borders are done by selecting the top and bottom most ridges within the left and right borders, respectively. The upper and lower borders are also in Figure 4(b) in red color, where the lower border is correct, but upper border is incorrect with respect to the non-text content area of the page.



(a) left and right borders (blue color) (b) initial upper and lower borders (red color) (c) dragged and extended upper and lower borders (d) page frame

Figure 4. Page Frame Detection: (a) left and right borders (blue colors) are detected using starting and ending points of ridges (green color), (b) the top most and the bottom most ridges inside vertical borders are selected as upper and lower borders; non-text parts (black color), that were deleted in preprocessing, are pasted back into the document image, (c) the upper and lower borders are dragged up to the top most pixel and bottom most pixel of the non-text elements, and finally both of them are extended upg to the page width, (d) page frame

The initial page frame possesses only non-text elements which lie between text-lines and misses others, as shown in Figure 4(b). The page frame is improved by dragging the upper and/or lower borders according to the non-text elements such that: i) if the top most pixel of non-text elements is above the top most pixel of the upper border, the upper border is dragged up to the top most pixel of non-text element, ii) similarly, if the bottom most pixel of non-text elements is below the bottom most pixel of the lower border, the lower border is dragged up to the bottom most pixel of non-text element. Finally, the upper and lower borders are extended across the document width using polynomial fitting. The upper and lower borders after dragging and extending are shown in Figure 4(c). The final page frame is shown in Figure 4(d). Some of example results of the presented page frame detection method on DFKI-I dataset are also shown in Figure 5.

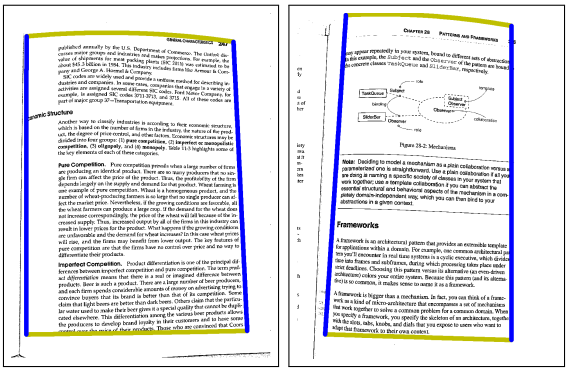


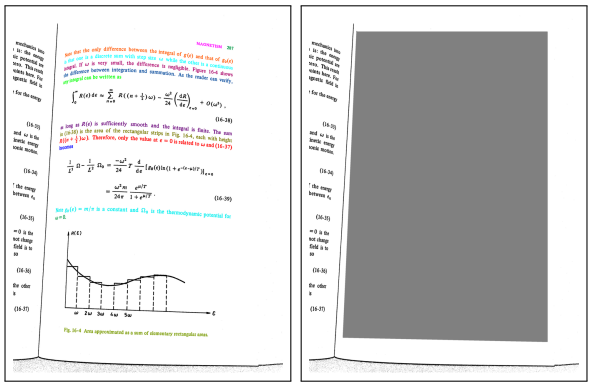
Figure 5. Sample results of our page frame detection method for DFKI-I dataset.

Generally, the starting points of some of the text-lines in a document image coincide with the document's left border line and similarly the ending points of some of the text-lines coincide with the right border line. The ridge alignment step helps in propagating this information to neighboring text-lines. Therefore, left and right borders estimation using starting and ending points of text-lines gives correct results In a special case where a document image contains only short or centered text-lines with non-text elements spanning throughout the page width, the left and right borders can not estimate the actual page contents area. In order to solve this problem, the left and/or right borders can also be dragged with respect to non-text elements, same as it is done in case of upper and/or lower border dragging.

III. Experiments and Results

We have compared our page frame detection method with state-of-the-art methods [8], [11] by evaluating them on publicly available DFKI-I (CBDAR 2007 dewarping contest) dataset [16]. We have conducted two different experiments for performance evaluation: i) text-line based evaluation, ii) pixel based evaluation.

DFKI-I dataset contains 102 grayscale and binarized document images of pages from several technical books captured using an off-the-shelf hand-held digital camera in a normal office environment. Document images in this dataset consist of warped text-lines with a high degree of curl, different directions of curl within an image, non-text (graphics, halftone, etc.) components, and a lot of textual and non-textual border noise. Together with ASCII-text ground-truth, this dataset also contains pixel based ground-truth for zones, text-lines, formulas, tables and figures. For text-line based performance evaluation method, text-line based ground-truth images are generated from the original ground-truth images. A text-lines based ground-truth image contains labeling only for text-lines and all the other foreground



(a) labeled text-lines (b) masked page contents

Figure 6. Text-line based and page contents based ground-truth images for an example image in DFKI-I (CBDAR 2007 dewarping contest) dataset [16].

objects, like formulas, tables and figures, are marked as noise with black color. For pixel based performance evaluation method, ground-truth images are generated by masking the actual page contents only. An example image and its corresponding text-lines and pixel based ground-truth images are shown in Figure 6.

In document images, text-lines are the main source of information from optical character recognition (OCR) point of view. For each text-line in a text-line based ground-truth image, the pixel-correspondence (P) is defined as the ratio of the number of overlapping pixels between the ground-truth image and the correspoding cleaned image and total number of pixels of a particular text-line. Text-line based performance evaluation metrics using pixel-correspondence (P) were defined in [16]. Here, we use the same metrics. These metrics are defined as follows: i) TI: totally-in text-line ($P \geq 90\%$), ii) TO: totally-out text-line ($P = 0\%$), and iii) PI: partially-in text-line ($P < 90\%$). These metrics measures the percentage of totally-in, partially-in, and totally-out text-lines within page contents with respect to the page frame. The text-line based performance evaluation of the proposed page frame detection method, Shafait et. al [8], and Stamatopoulos et. al [11] page frame detection methods are shown in Table I. The results show that our method outperforms other two methods mentioned above.

Tex-line based performance metrics only measure the performance of a page frame detection method for text within actual page content area. They report nothing about the performance of a page frame detection method for marginal noise as well as non-text elements within page content area. Furthermore, text-line based performance evaluation is not a useful measure for the case where the boundary of a complete document image, which contains both textual and non-textual noise, is marked as the page frame. In such a case, text-line based performance evaluation reports 100% totally-in text-lines with no partial-in or totally-out text-lines.

Table I
TEXT-LINE BASED PERFORMANCE EVALUATION OF OUR PAGE FRAME DETECTION METHOD AND STATE-OF-THE-ART METHODS [8], [11] ON DFKI-I DATASET. THE RESULTS OF SHAFAIT ET. AL METHOD [8] HAVE BEEN COPIED FROM THEIR PAPER. TI: TOTALLY-IN TEXT-LINES; TO: TOTALLY-OUT TEXT-LINES; PI: PARTIALLY-IN TEXT-LINES. (NOTE: TOTAL NUMBER OF DOCUMENT IMAGES = 102; TOTAL NUMBER OF TEXT-LINES = 3097.)

Method	TI	PI	TO
Shafait et. al [8]	95.6%	2.3%	2.1%
Stamatopoulos et. al [11]	96.48%	0.71%	2.81%
our method	**98.10%**	**1.13%**	**0.78%**

Therefore, text-line based performance evaluation alone is not enough for comparing the performance of different page frame detection algorithms. In order to measure how well a page frame detection method works with respect to both marginal noise and actual page contents, a pixel based performance evaluation is used. Our pixel based performance evaluation method measures the pixel-correspondence (P) for both actual page contents and marginal noise between a ground-truth image and the correspoding cleaned image. Pixel correspondence for page content is defined as the ratio of the number of overlapping pixels between the page contents of ground-truth image and the correspoding cleaned image and total number of page contents pixels in ground-truth image. Likewise, the pixel correspondence is defined for marginal noise. The pixel based performance evaluation results of our proposed method and Stamatopoulos et. al [11] page frame detection method are shown in Table II. It shows that both methods give good performance for actual page contents, but our method performs better for marginal noise cleanup.

Table II
PIXEL BASED PERFORMANCE EVALUATION OF OUR PAGE FRAME DETECTION METHOD AND STATE-OF-THE-ART METHOD [11] ON DFKI-I DATASET. 'PAGE CONTENTS' REPRESENTS THE PERCENTAGE OF PAGE CONTENTS INSIDE DETECTED PAGE FRAME. 'MARGINAL NOISE' REPRESENT THE PERCENTAGE PF NOISE OUTSIDE DETECTED PAGE FRAME. (NOTE: TOTAL NUMBER OF PAGE CONTENTS PIXELS = 48188808 (88.52%); TOTAL NUMBER MARGINAL NOISE PIXELS = 6247054 (11.48%)).

method	Page Contents	Noise
Stamatopoulos et. al [11]	99.11%	36.04%
our method	98.96%	74.81%

IV. DISCUSSION

In this paper, we have presented a page frame detection method for warped camera-captured document images. Our method uses text-lines and non-text contents information

for detecting page frame (left, right, upper, and lower borders). We have developed a ridge based text-line finding method [12] and a multiresolution based text/non-text segmentation method [14], which we have used here for detecting text-lines and non-text elements, respectively. For the performance evaluation of the presented method and its comparison with state-of-the-art methods, two different methodologies, text-line based and pixel based, have been used. For both performance evaluation methodologies, the presented method has achieved better results than Shafait et. al [8] and Stamatopoulos et. al [11] page frame detection methods, as shown in Table I and Table II.

REFERENCES

[1] F. Shafait and T. M. Breuel, "The effect of border noise on the performance of projection based page segmentation methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 4, pp. 846–851, 2011.

[2] D. X. Le, G. R. Thoma, and H. Wechsler, "Automated borders detection and adaptive segmentation for binary document images," in *13th Int. Conf. on Pattern Recognition*, Vienna, Austria, Aug. 1996, pp. 737–741.

[3] B. T. Avila and R. D. Lins, "Efficient removal of noisy borders from monochromatic documents," in *Int. Conf. on Image Analysis and Recognition*, Porto, Portugal, Sep. 2004, pp. 249–256.

[4] K. C. Fan, Y. K. Wang, and T. R. Lay, "Marginal noise removal of document images," *Pattern Recognition*, vol. 35, no. 11, pp. 2593–2611, 2002.

[5] L. Cinque, S. Levialdi, L. Lombardi, and S. Tanimoto, "Segmentation of page images having artifacts of photocopying and scanning," *Pattern Recognition*, vol. 35, no. 5, pp. 1167–1177, 2002.

[6] F. Shafait and T. M. Breuel, "A simple and effective approach for border noise removal from document images," in *13th IEEE Int. Multi-topic Conference*, Islamabad, Pakistan, Dec 2009.

[7] F. Shafait, D. Keysers, and T. M. Breuel, "Efficient implementation of local adaptive thresholding techniques using integral images," in *Proc. SPIE Document Recognition and Retrieval XV*, San Jose, CA, USA, Jan. 2008, pp. 101–106.

[8] F. Shafait, J. van Beusekom, D. Keysers, and T. Breuel, "Document cleanup using page frame detection," *International Journal on Document Analysis and Recognition*, vol. 11, pp. 81–96, 2008.

[9] N. Stamatopoulos, B. Gatos, and T. Georgiou, "Page frame detection for double page document images," in *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, Boston, MA, USA, 2010, pp. 401– 408.

[10] H. Fan, L. Zhu, and Y. Tang, "Skew detection in document images based on rectangular active contour," *International Journal on Document Analysis and Recognition*, vol. 13, no. 4, pp. 261–269, 2010.

[11] N. Stamatopoulos, B. Gatos, and A. Kesidis, "Automatic borders detection of camera document images," in *Proceedings of Second International Workshop on Camera-Based Document Analysis and Recognition*, Curitiba, Brazil, 2007, pp. 71–78.

[12] S. S. Bukhari, F. Shafait, and T. M. Breuel, "Ridges based curled textline region detection from grayscale camera-captured document images," in *International conference on Computer Analysis of Images and Patterns*, ser. Lecture Notes in Computer Science, Muenster, Germany, 2009, vol. 5702, pp. 173–180.

[13] A. Ulges, C. Lampert, and T. Breuel, "Document image dewarping using robust estimation of curled text lines," in *Proc. Eighth Int. Conf. on Document Analysis and Recognition*, Aug. 2005, pp. 1001–1005.

[14] S. S. Bukhari, F. Shafait, and T. M. Breuel, "Improved document image segmentation algorithm using multiresolution morphology," in *Proc. SPIE Document Recognition and Retrieval XVIII*, San Jose, CA, USA, Jan. 2011.

[15] ——, "Dewarping of document images using coupled-snakes," in *Proceedings of Third International Workshop on Camera-Based Document Analysis and Recognition*, Barcelona, Spain, 2009, pp. 34–41.

[16] F. Shafait and T. M. Breuel, "Document image dewarping contest," in *2nd Int. Workshop on Camera-Based Document Analysis and Recognition*, Curitiba, Brazil, Sep. 2007, pp. 181–188.