Goal-Oriented Rectification of Camera-Based Document Images

Nikolaos Stamatopoulos, Basilis Gatos, Ioannis Pratikakis, *Member, IEEE*, and Stavros J. Perantonis

***Abstract*—Document digitization with either flatbed scanners or camera-based systems results in document images which often suffer from warping and perspective distortions that deteriorate the performance of current OCR approaches. In this paper, we present a goal-oriented rectification methodology to compensate for undesirable document image distortions aiming to improve the OCR result. Our approach relies upon a coarse-to-fine strategy. First, a coarse rectification is accomplished with the aid of a computationally low cost transformation which addresses the projection of a curved surface to a 2-D rectangular area. The projection of the curved surface on the plane is guided only by the textual content's appearance in the document image while incorporating a transformation which does not depend on specific model primitives or camera setup parameters. Second, pose normalization is applied on the word level aiming to restore all the local distortions of the document image. Experimental results on various document images with a variety of distortions demonstrate the robustness and effectiveness of the proposed rectification methodology using a consistent evaluation methodology that encounters OCR accuracy and a newly introduced measure using a semi-automatic procedure.**

***Index Terms*—Document image analysis, document image processing, document image rectification, image dewarping.**

I. Introduction

DOCUMENT image acquisition by a flatbed scanner or a digital camera often results in several unavoidable image distortions (see Fig. 1) due to the form of printed material (e.g., bounded volumes), the camera setup or environmental conditions (e.g., humidity that causes page shrinking). Text distortions not only reduce document readability but also affect the performance of subsequent processing such as document layout analysis and optical character recognition (OCR).

Manuscript received January 12, 2010; revised July 13, 2010; accepted September 03, 2010. Date of publication September 27, 2010; date of current version March 18, 2011. This work was supported in part by the European Community's Seventh Framework Programme under Grant No. 215064 (project IMPACT). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Mark (Hong-Yuan) Liao.

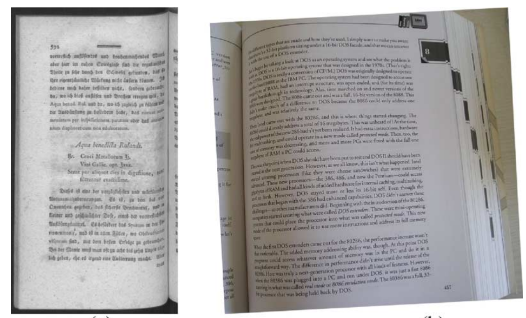
N. Stamatopoulos is with the Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, Greece and the Institute of Informatics and Telecommunications, National Center for Scientific Research "Demokritos," Athens GR-15310, Greece (e-mail: nstam@iit.demokritos.gr).

B. Gatos and S. J. Perantonis are with the Institute of Informatics and Telecommunications, National Center for Scientific Research "Demokritos," Athens GR-15310, Greece (e-mail: bgat@iit.demokritos.gr; sper@iit.demokritos.gr).

I. Pratikakis is with the Department of Electrical and Computer Engineering, Democritus University of Thrace, GR-67100 Xanthi, Greece (e-mail: ipratika@ee.duth.gr)

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2010.2080280



(a) (b)

Fig. 1. Examples of document images captured by (a) flatbed scanner (b) digital camera.

Over the last decade, many different techniques have been proposed for document image rectification [1] that can be classified into two main categories based on 1) 3-D document shape reconstruction [2]–[9] and 2) 2-D document image processing [10]–[24]. Techniques of the former category obtain the 3-D information of the document image using special setup or reconstruct the 3-D model from information existing in document images. On the other hand, techniques in the latter category do not depend on auxiliary hardware or prior information but they only rely on 2-D information.

In this paper, we propose a goal-oriented rectification methodology to compensate for undesirable distortions of document images captured by flatbed scanners or hand-held digital cameras. The proposed technique is directly applied to the 2-D image space without any dependence to auxiliary hardware or prior information. It first detects words and text lines to rectify the document image in a coarse scale and then further normalize individual words in finer detail using baseline correction. Although during the coarse rectification stage word and text line detection is applied at the original distorted document image, which is a well-known hard task, potential erroneous detection results do not seriously affect this stage as only some specific points are required. Experimental results on several document images with a variety of distortions show that the proposed method produces rectified images that give a significant boost in OCR performance. The flowchart of the proposed rectification methodology is shown in Fig. 2.

This work is an extension of [10] and [11] which incorporates a new method for the curved surface projection, the word baseline fitting as well as the restoration of horizontal alignment. We also propose to rectify the distortion of individual words using baseline estimation. Finally, we use evaluation method in [25] based on matching manually marked points of the original

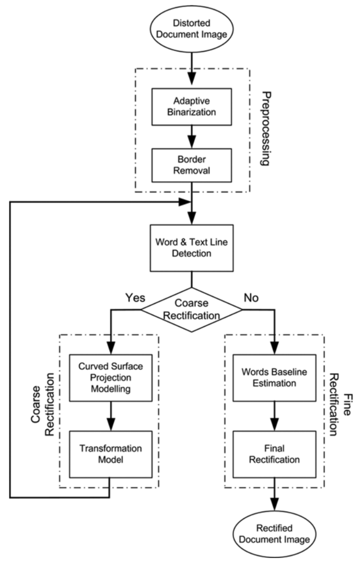


Fig. 2. Flowchart of the proposed coarse-to-fine rectification methodology.

image and corresponding points of the rectified image. A quantitative measure is calculated to evaluate the performance of our method.

The remainder of the paper is organized as follows. In Section II the related work including its main drawbacks is discussed. Section III focuses on the proposed methodology. First, a general overview and the contribution of the proposed work is given. Then, a detailed analysis of the steps involved in the proposed methodology is provided. In particular, we detailed the preprocessing as well as word and text line detection stages. Finally, the coarse-to-fine rectification methodology is given. Experimental results indicating the performance of the proposed methodology are discussed in Section IV while conclusions and remarks on future directions are drawn in Section V.

II. Related Work

In this section, we present all major methodologies that address the rectification problem. First, we present the rectification techniques that are based on 3-D document shape reconstruction and second, we present the techniques that are based on 2-D document image processing. Rectification techniques of the former category can be further divided into techniques that take into account information using special equipment and those techniques in which this additional source of information is not used. Rectification techniques that are based on 2-D document image processing can be further divided into two subcategories. The first involves the techniques that are based on the detection of distorted text lines at the original document image. On the other hand, the techniques that belong to the second subcategory do not rely on the detection of distorted text lines but they aim to find spatial transformations between the warped and dewarped document images by analyzing the 2-D content such as document boundaries or known reference points.

A. Rectification Techniques Based on 3-D Document Shape Reconstruction

In this category, rectification techniques rely upon extraction of the 3-D information of the document and they can be further divided into two subcategories. Techniques of the first subcategory obtain the 3-D shape of the document image using special equipment such as laser scanners [2], stereo cameras [3], [4], or structured light setups [5]. The dependence on special equipment prevents these techniques from being used in an unconstrained environment. On the other hand, techniques of the second subcategory reconstruct the 3-D model from information existing in document images. Cao *et al.* [6] propose a method to rectify warping distortions in document images by constructing a cylinder model. Apart from the cylinder shape assumption, they also have a limitation on the pose that requires the image plane to be parallel to the generatrix of the page cylinder. Liang *et al.* [7] model the page surface by curved developable surfaces to estimate the 3-D shape of the page using texture flow fields. This method is based on the assumptions that the document is either flat or smoothly curved and the camera is a standard pinhole camera. Finally, Tan *et al.* [8] and L. Zhang *et al.* [9] use a shape-from-shading formulation to reconstruct the 3-D shape of the document's surface. These techniques require knowledge of lighting, which in most of the cases, is unknown.

B. Rectification Techniques Based on 2-D Document Image Processing

In this category, rectification techniques rely on the use of 2-D information available in document images. The majority of these rectification techniques [12]–[22] are based on the detection of distorted text lines at the original document image which is a well-known hard task. Some of these techniques propose a method to straighten distorted text lines by fitting a model to each text line. Lavialle *et al.* [12] use an analytical model with cubic B-splines, Wu and Agam [13] use a non-linear curve for each text line, L. Zhang and Tan [14] represent the text lines using natural cubic splines, Ezaki *et al.* [15] use cubic splines not only to model the distorted text lines but also the space between them while in [16] each distorted text line is polynomially approximated. All above mentioned techniques suffer from various limitations. Specifically, the approach in [12] is not efficient in the case of inhomogeneous line spacing and method [13] is based on several heuristics, while it requires that the user should interactively specify the four corner points of the warped image which is not practical and cannot handle non-uniform columns in the target mesh, as well. In [14], L. Zhang and Tan assume that the book spine is found along iso-parametric lines and method in [15] uses complex computations along with a line-warping model which is not so accurate. Finally, Mischke and Luther [16] require a pre-processing step to correct the skew of the warped document and confine the restoration to a fixed type of warping, making it hard to generalize.

A few more rectification techniques also rely on text line detection but they emphasize on baseline finding. Ulges *et al.* [17] estimate quadrilateral cell for each letter based on local baselines finding and then map to a rectangle of corrected size and position in the dewarped image. Their method is not generic since it is based on the assumption that the original page contains only straight lines that are approximately equally spaced and sized while spacing between words is not large. Lu *et al.* [18] restore the document by dividing images into multiple quadrilateral patches based on the exploitation of the vertical stroke boundaries (VSBs) and text baselines. This method is based on several heuristics and is limited on documents printed in Latin languages. In [19], Schneider *et al.* use local orientation features extracted by text line baselines to interpolate a vector field from which a warping mesh is derived. A drawback of this approach is that it is hard to define such characteristic points of transitions so that stable approximation of baselines is achieved. Bukhari *et al.* [20] map characters over each curled baseline pair (upper and lower) to its corresponding straight baseline pair. This method is sensitive to large and various distortions, especially among the same text line. Fu *et al.* [21] assume that the image surface is a cylinder and generate a transformation to flatten the document image. The main disadvantage of this method is that it requires complex computation and, therefore, is time-consuming, while the assumption that a single cylinder fits to a deformed page is not generic. Finally, Y. Zhang *et al.* [22] take a rough text line and character segmentation to estimate the warping direction. Then, a Thin-Plate Splines (TPS) interpolation is used to restore the image. Text line and character segmentation using projections at the original warped document can cause many segmentation errors.

There are also rectification techniques that do not rely on the detection of distorted text lines but they aim to find spatial transformations between the warped and dewarped document images by analyzing the 2-D content such as document boundaries or known reference points. Brown and Tsoi [23] use document boundary interpolation to correct geometric distortions and shading artifacts present in images of art-like materials. They use a physical pattern to guide the uniform parameterization, so it is limited to some specific document distortions. Masalovitch and Mestetskiy [24] approximate deformation of interlinear spaces in an image based on elements of image's skeleton that lie between the text lines. This method is sensitive to the approximation of vertical borders deformation in text blocks, which diminish accuracy.

III. Proposed Methodology

A. Overview

The proposed rectification methodology uses only 2-D information from document images without any dependence on auxiliary hardware or prior knowledge. It adopts a coarse-to-fine rectification strategy. The coarse rectification step aims to restore the large distortions of the document images and the fine rectification step aims to restore the local distortions and achieve an optimal rectification of the document image. Before we proceed with the coarse-to-fine rectification process, we apply a preprocessing step and detect words and text lines. Word and text line detection is applied not only before the coarse rectification step but also before the fine rectification step. In coarse rectification step, only some specific points are required, therefore, potential erroneous detection results do not seriously affect this stage. Next, in the fine rectification step, text line detection is applied at the coarse rectified image, thus having improved initial conditions that can lead to successful detection results. Furthermore, in contrast to state of the art techniques [14]–[16], [21], and [23], which are based on specific patterns or camera setup parameters making them hard to generalize, or use complex computations, the proposed coarse rectification stage is accomplished with the aid of a computationally low cost transformation which is not based on specific model primitives or camera setup parameters, so it is rendered more generic. Although the proposed rectification methodology requires that the text content of the document image should be justified and it should not contain cursive handwritten text in order to be able to detect the words, it is independent of the document's language and it can deal with documents which contain inhomogeneous text line spacing as well as non-text content like pictures, graphs, etc. It is worth to note that the proposed rectification method processes only single-column document images. Therefore, in the case that it is required to process document images that contain two pages or multi columns, it is considered that a method which should split the pages or detect the columns of the document, such as method presented in [26], should have already been applied.

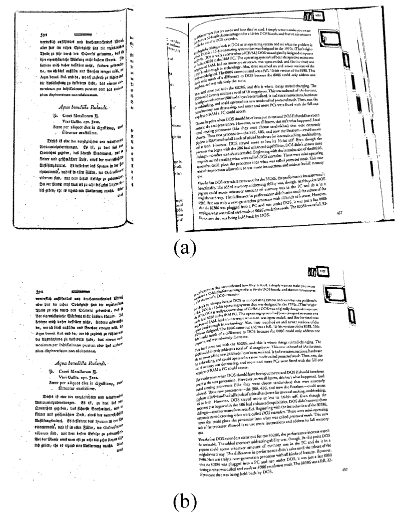


Fig. 3. Preprocessing results of document images shown in Fig. 1; (a) adaptive binarization; (b) border removal.

B. Preprocessing

Before we proceed with the coarse-to-fine rectification process, we apply a preprocessing step at the original distorted document image which consists of an adaptive binarization using the technique proposed in [27] as well as black and text border removal based on [28]. Fig. 3 shows the resulting images after applying these steps to the distorted document images shown in Fig. 1.

C. Word and Text Line Detection

In this step, a word and text line detection technique for distorted document images is introduced. In the overall rectification process, word and text line detection is applied not only before the coarse rectification step but also before the fine rectification step. However, each step requires a different level of tolerance in erroneous detection results. The coarse rectification stage applies this step at the original distorted document image, so there may be some detection errors, especially when the distortions are relatively large. However, at this stage, we do not care whether the text line detection is accurate, since we just need some specific points in order to model the curved surface projection on the plane and we will not use each detected text line to correct the distortions of the document. For the sake of clarity, we will report on some possible errors of the detection step. In the case that the proposed method splits or merges words of the same text line this does not influence the process of text line detection. On the other hand, if the methodology merges words of adjacent text lines it will result at erroneous text line detection. However, these errors will not influence the coarse rectification stage since we need only the start and end points of each text line while the short text lines, which might have been produced by these errors, are eliminated (see Section III-D). Finally, if the method could not detect a word, it is possible to split a text line into two text lines which will be also eliminated. In fine rectification, word and text line detection is applied at the coarse rectified document image thus having improved initial conditions that can lead to successful detection results.

Before we proceed with the word and text line detection we estimate the dominant character height (AH) in order to temporarily remove the components like pictures graphs, noisy, etc and keep only the text which will be used in the following steps. In this way, large and small connected components are removed, so even if a few non-text elements have been remained they will be treated as words. All parameters used in this stage depend on the dominant character height, so for each document image a proper adaptation is applied. For the calculation of the dominant character height we apply connected component labeling and then we calculate the height of each bounding box of the connected components that results in constructing height histogram. The dominant character height is denoted as the maximum value of the histogram. So, we remove the connected components which satisfy the following condition:

$$h > 3 * AH \ \text{ or } \ h < \frac{AH}{4} \ \text{ or } \ \mathrm{w} < \frac{AH}{4} \qquad (1)$$

where h and w denote connected component's height and width, respectively.

For the word detection we first apply a horizontal smoothing using the Run Length Smoothing Algorithm (RLSA) [29] with threshold $Th = 0.5{*}AH$ (see Fig. 4(b)), followed by a connected component labeling in order to detect words (see Fig. 4(c)). Once the words have been detected, we proceed with text line detection. In a left-right and top-down scanning, we detect the first word and assign this as the first word of the first text line. Following that, horizontally neighboring words, in left and right direction, are consecutively linked in order to detect all the words of the first text line. Finally, we continue the scanning and repeat this process for the remaining text lines until all the words are assigned to text lines (see Fig. 4(d)).

The process of linking two neighboring words is addressed as follows: Let (x_1, y_1), (x_2, y_2) denote the bounding box coordinates of an assigned word and (x_1', y_1'), (x_2', y_2') denote the bounding box coordinates of a candidate neighbor word. From all words in the right side of the assigned word which satisfy the condition $[y_1, y_2] \cap [y_1', y_2'] \neq \emptyset$ (represents the horizontal overlapping), we select the one with the smaller distance $D = x_1' - x_2$ only if $0 < D < 6{*}AH$. Since many words may satisfy the condition of horizontal overlapping, selecting the one with the smaller distance, we select the immediate neighbor word of the same text line. Next, this word is assigned as processed and we search in the right side for a neighbor word till the last word of the text line is assigned as processed. A similar treatment is applied in the left side. In order to examine how this stage influences the proposed rectification methodology, in Section IV, we present experimental results on the performance evaluation of the proposed word detection technique against state of the art techniques.

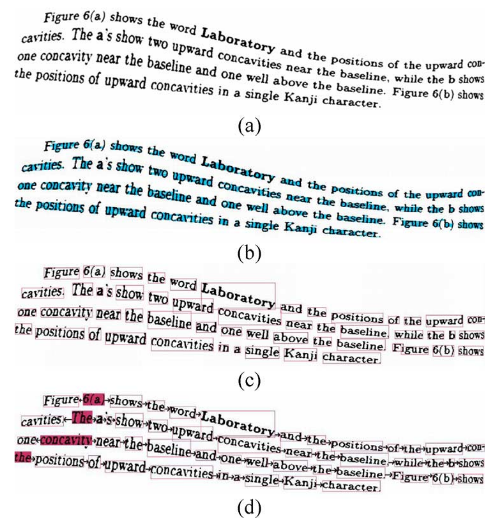


Fig. 4. Example of word and text line detection; (a) original distorted document image; (b) result after horizontal smoothing; (c) detected words; (d) detected text lines; the first word of each text line detected in the left-right and top-down scanning is indicated with a filled box and the arrows show the linked neighbor words of the same text line.

D. Coarse-to-Fine Rectification

The core of the proposed rectification method is built with a coarse-to-fine strategy. The coarse rectification step aims to restore the large distortions of the document images, so that a rough rectification should be achieved. The rectified outcome will be given as an input in the next step (fine rectification) and help it to produce more accurate results in word and text line detection stage. On the other hand, the fine rectification step aims to restore the local distortions of the document image and achieve an optimal rectification of the document image. A detailed analysis of both steps is described in the sequel.

1) Coarse Rectification: In this step, we apply a computationally low cost transformation which addresses the projection of a curved surface to a 2-D rectangular area in order to achieve a coarse rectification of the document image. Compared with state of the art techniques [14], [21], and [23], which also use boundaries to delimit the required dewarping, our approach does not use any physical pattern or global model (e.g., cylinder) for modelling the distortion of the document image. Methods [14] and [23] use particular types of interpolation based on Gordon

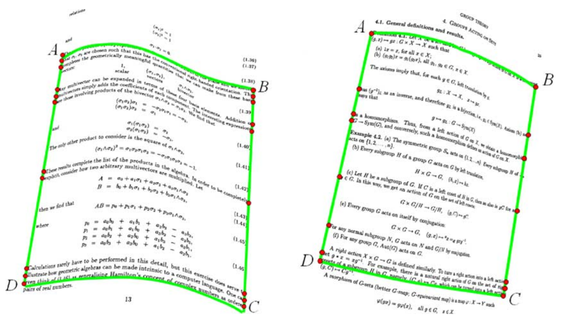


Fig. 5. Examples of modelling the curved surface projection on the plane; Start and end points which participate in the procedure are indicated with circles.

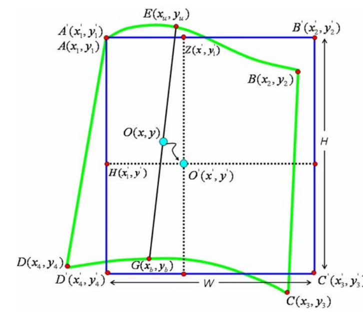


Fig. 6. Transformation model in coarse rectification: $W = |\widehat{AB}|$, $H = |AD|$ and $A'(x_1', y_1') \equiv A(x_1, y_1)$.

surface model [30] and bilinearly blended Coons [31], respectively. In our method, we create a correspondence between the points of the two curved line segments at the top and bottom area, upon which the mapping from the projection of a surface to a rectangle is applied. The distinct steps of coarse rectification stage are explained in the following sections.

Modelling the Curved Surface Projection on the Plane: Once the text lines have been detected, we proceed in modelling the projection of the curved surface. We consider that the projected result is delimited by the two curved lines which fit the top and bottom text lines along with the two straight lines which fit the left and right text boundaries. Let A, B, C, and D denote the dominant corner points of the projection of the curved surface (see Fig. 5).

First, the straight lines AD and BC which correspond to the left and right text boundaries are estimated. The start and end points of each text line are detected and the short text lines are excluded using the average length of all text lines. In this way, short text lines such as titles, marginal text, math types, etc., are eliminated and we retain the most representative text lines. Among these text lines remained, some of them do not start (or end) from the beginning (or ending) of the document frame, mainly the first and last text line of a paragraph. So, in order to further eliminate these text lines an iterative procedure is applied. If the deviation of the estimated straight line is greater than the dominant character height we exclude the start (or end) point with the maximum distance and recalculate the straight line. This iteration stops when the above criterion is satisfied or two text lines have been remained (see Fig. 5). Since the iterative procedure uses only long text lines, the potential deviation of the estimated straight line will not be large. Therefore, a worst case scenario may lead to a less precise modelling of the curved surface projection, that will consequently lead to a less successful coarse rectification result, but the method will not fail.

Next, the curved lines AB and DC which correspond to the top and bottom text lines are estimated. In order to select appropriate text lines of the document with representative deformation we select the top and bottom text lines which participate in the calculation of the straight lines AD and BC (see Fig. 5). In this way, short text lines are excluded. Using the upper and bottom points of the selected text lines respectively, the coefficients of 3rd degree polynomial are calculated. Appendix A details the estimation procedure of the straight and curved line segments.

Transformation Model: After modelling the projection of the curved surface on the plane delimited by the curved line segments AB and DC along with the straight line segments AD and BC, our goal is to generate a transformation that maps the projected curved surface to a 2-D rectangular area. Let A', B', C' and D' denote the corner points of the rectangular area (see Fig. 6). Also, let $\left|\widehat{AB}\right|$ and $|AB|$ represent the arc length and the Euclidean distance, respectively, between points A and B.

First, we locate the corner points A', B', C' and D' of the rectangular area. One of them coincides with one of the dominant corner points of the projection of the curved surface ($A'(x_1', y_1') \equiv A(x_1, y_1)$) and the rest of them are calculated by taking into account the width W and the height H of the rectangular area (see Fig. 6)

$$W = \min(|\widehat{AB}|, |\widehat{DC}|) \ \&\ H = \min(|AD|, |BC|). \tag{2}$$

Once the rectangular area $A'B'C'D'$ has been defined, each point $O(x, y)$ in the projection of the curved surface is mapped to the corresponding point $O'(x', y')$ in the rectangular area. Each point $O(x, y)$ is defined by two points (E and G) according to the arc length of the top and bottom curved line segments (see Fig. 6). Points E and G satisfy the following condition:

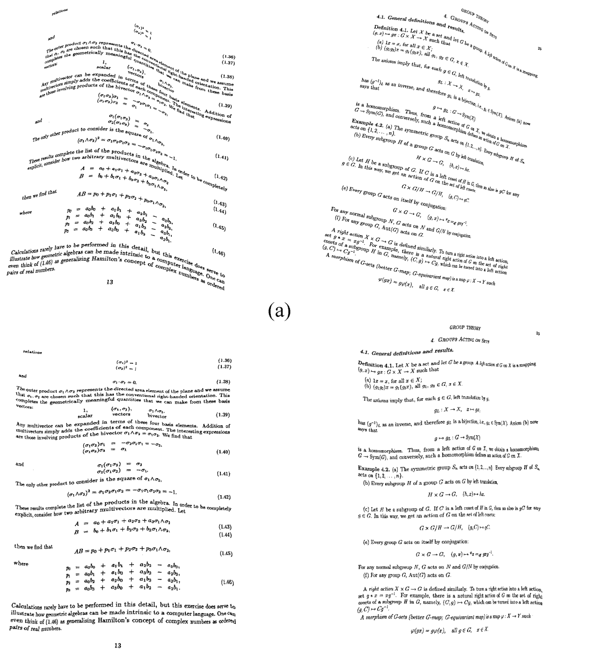
$$\frac{|\widehat{AE}|}{|\widehat{AB}|} = \frac{|\widehat{DG}|}{|\widehat{DC}|}. \tag{3}$$

The corresponding point $O'(x', y')$ in the rectangular area is calculated by preserving the ratio between the projection of the curved surface and the rectangular area in the x direction (4) as well as in the y direction (5). Based on this goal, the points $Z(x', y_1')$ and $H(x_1', y')$ are calculated using the following equations:

$$\frac{|\widehat{AB}|}{|\widehat{AE}|} = \frac{W}{|A'Z|} \Rightarrow |A'Z| = \frac{W}{|\widehat{AB}|}|\widehat{AE}| \tag{4}$$

$$\frac{|EG|}{|EO|} = \frac{H}{|A'H|} \Rightarrow |A'H| = \frac{H}{|EG|}|EO|. \tag{5}$$

The detailed description of the transformation model is given in Appendix B. Examples of coarse rectification step are shown



(a)

(b)

Fig. 7. Examples of coarse rectification step: (a) original document images; (b) corresponding rectified document images.

in Fig. 7. As we can observe, the coarse rectification step restores the large distortions of the document images and achieve a rough restoration of them. Furthermore, the vertical alignment of the documents is corrected.

2) *Fine Rectification:* Before we proceed with the fine rectification, word and text line detection is applied at the coarse rectified document image. In the methodology's flowchart (see Fig. 2), the distinct steps of fine rectification stage can be seen. The remainder steps followed the procedure are explained in the following sections.

Word Baseline Fitting: In this step, the lower and upper baselines are detected which delimit the main body of the words. Starting from the smoothed image after applied RLSA, we follow the methodology given in [32] which is used for lower baseline detection. According to this approach, a linear regression is applied on the set of points that are the lowest foreground pixels for each column of the word. A similar procedure is used to calculate the upper baseline. After this procedure, upper and lower baselines of word W_i are defined as:

$$y = a_i^u x + b_i^u \ \& \ y = a_i^l x + b_i^l. \tag{6}$$

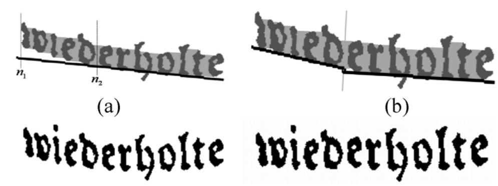
Furthermore, upper and lower baseline slopes θ_i^u and θ_i^l of word W_i are denoted as:

$$\theta_i^u = \arctan(a_i^u) \ \& \ \theta_i^l = \arctan(a_i^l) \tag{7}$$

and upper and lower baseline deviation S_i^u and S_i^l are denoted as:

$$S_i^u = \sum_{j=1}^{n} |y_j^u - (a_i^u x_j + b_i^u)| \ \& \ S_i^l = \sum_{j=1}^{n} |y_j^l - (a_i^l x_j + b_i^l)| \tag{8}$$

Fig. 8. Examples of upper and lower baseline estimation; the selected baseline according to (9) is in black.



(a) (b)

(c) (d)

Fig. 9. Example of baseline estimation and rotation of a distorted word; (a) initial baseline estimation: n_1 and n_2 indicate the interval in which the baseline deviates from the word (see (10)); (b) baseline estimation after word has been splitted; (c)-(d) rotation of the word using the baseline of (a) and (b), respectively.

where n is the number of columns of word W_i, (x_j, y_j^u) and (x_j, y_j^l) denote the position of the uppermost and lowest foreground pixel in column j of word W_i.

Each word W_i should be specified only from one baseline $(a_i, b_i, \theta_i, S_i)$. Since the smaller absolute baseline slope and deviation is usually the most representative, the baseline of word W_i is defined as

$$(a_i, b_i, \theta_i, S_i) = \begin{cases} (a_i^u, b_i^u, \theta_i^u, S_i^u), & \text{if } (|\theta_i^u| < |\theta_i^l|) \text{ and } (S_i^u < S_i^l) \\ (a_i^l, b_i^l, \theta_i^l, S_i^l), & \text{otherwise} \end{cases}. \tag{9}$$

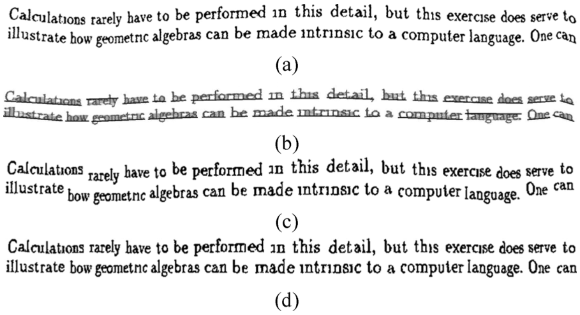
Examples of baseline estimation are given in Fig. 8.

Several times, words suffer from distortions, so the appropriate baseline cannot be estimated (see Fig. 9(a)). Consequently, only the rotation and translation of the words at the next step would not be enough in order to restore them (see Fig. 9(c)). For this reason, we iteratively split the word W_i and process each part of it independently if the word W_i satisfies the following criterion:

$$\exists\, n_1, n_2 : \prod_{j=n_1}^{n_2} |y_j - (a_i x_j + b_i)| \neq 0 \text{ and } (n_2 - n_1) \geq 2 * AH \tag{10}$$

where (x_j, y_j) is the position of the uppermost or lowest foreground pixel in column j of word W_i. According to this criterion, we split a word when the baseline deviates from the word for an interval larger than $2{*}AH$ (twice the dominant character height) (see Fig. 9(a)). It indicates that a precise baseline cannot be estimated; hence the word may suffer from distortions. For this reason, we split the word and estimate the baseline of each part separately (see Fig. 9(b)). At the next step, each part of the word will be rotated and translated and the distorted word will be totally corrected (see Fig. 9(d)).

Final Rectification: This is the final step of the proposed methodology where all detected words $W_i(x, y)$ are rotated and translated in order to obtain the final rectified document image. First, we proceed with the rotation of the words. Every word is rotated according to the word's baseline slope $(-\theta_i)$. After this process, every word is parallel in the y-direction; however, the horizontal alignment of the text lines is not justified (see


Calculations rarely have to be performed in this detail, but this exercise does serve to illustrate how geometric algebras can be made intrinsic to a computer language. One can

(a)

Calculations rarely have to be performed in this detail, but this exercise does serve to illustrate how geometric algebras can be made intrinsic to a computer language. One can

(b)

Calculations rarely have to be performed in this detail, but this exercise does serve to illustrate how geometric algebras can be made intrinsic to a computer language. One can

(c)

Calculations rarely have to be performed in this detail, but this exercise does serve to illustrate how geometric algebras can be made intrinsic to a computer language. One can

(d)

Fig. 10. Example of final rectification; (a) coarse rectified document image; (b) words baseline estimation; (c) rotation of the words according to (11); (d) translation of the words according to (12).

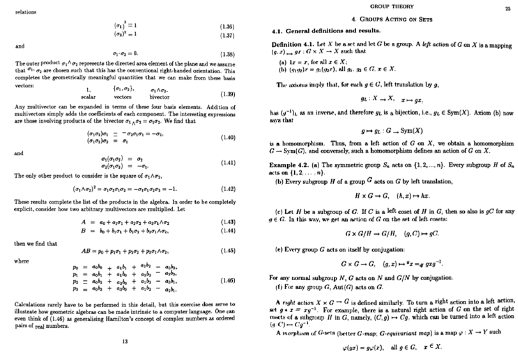


Fig. 11. Results after applying fine rectification step in the document images shown in Fig. 7(b).

Fig. 10(c)). The rotation of the word $W_i(x,y)$ is calculated as follows:

$$(x^r, y^r) = (x, (x - x_i^{\min}) * \sin(-\theta_i) + y * \cos(\theta_i)) \quad (11)$$

where $W_i^r(x^r, y^r)$ is the rotated word and $x_i^{\min}$ is the left side of the bounding box of the word $W_i(x,y)$. An example of rotation of a distorted word which has been split into two parts is given in Fig. 9.

After words rotation, all the words of every text line, expect from the leftmost or rightmost, must be vertically translated in order to restore horizontal alignment. The translation of the words is guided by the leftmost or rightmost word of every text line. In order to choose between them we calculate the average slope $A\theta_L$ of all leftmost words and the average slope $A\theta_R$ of all rightmost words of every text line and choose the side with the smaller average slope. With this process, we select to restore the horizontal alignment guided by the side of the document image with the minimum distortions, so we will achieve better rectification results.

Once the left or right side has been selected, the word reference points are detected. As word reference point we determine the point of the upper or lower baseline in the left side of the word after rotation. Concerning only the leftmost or rightmost word of every text line, we detect two reference points for each word according to the upper and lower baseline. The reason of having two reference points is that each word may be rotated either by its lower baseline or upper baseline slope (see (9)). Hence, it has to be translated so that its lower or upper reference point is aligned with the lower or upper reference point of the leftmost or rightmost word of the text line. The vertical translation of the word $W_i^r(x^r, y^r)$ is calculated as follows:

$$d_i = \begin{cases} y_0^{ur} - y_i^r, & \text{if } (a_i, b_i, \theta_i, S_i) \equiv (a_i^u, b_i^u, \theta_i^u, S_i^u) \\ y_0^{lr} - y_i^r, & \text{otherwise} \end{cases} \quad (12)$$

where the parameters $y_0^{ur} = (a_0^u x_0^{\min} + b_0^u) * \cos(\theta_0^u)$, $y_0^{lr} = (a_0^l x_0^{\min} + b_0^l) * \cos(\theta_0^l)$ correspond to the upper and lower word reference point of the leftmost or rightmost word and $y_i^r = (a_i x_i^{\min} + b_i) * \cos(\theta_i)$ correspond to the translated word reference point.

Once the rotation and translation of the word $W_i(x,y)$ have been calculated the final restoration of it is done as follows:

$$(x^{rt}, y^{rt}) = (x, y^r + d_i) \quad (13)$$

where $W_i^{rt}(x^{rt}, y^{rt})$ is the rotated and translated word (see Fig. 10).

Finally, we add all the components which have been removed. First, the co-ordinates (x_{cm}, y_{cm}) of the centre of mass of each component are calculated. Then, the centre of mass inherits the transformation factors (x^{rt}, y^{rt}) of the nearest pixel, which has been calculated according to (13). Then, we apply globally this transformation factor in all pixels each component so that the components will not be splitted. The result of applying fine rectification step in document images shown in Fig. 7(b) is given in Fig. 11.

IV. Experimental Results

To verify the validity of the proposed methodology we use as a performance measure the character and word accuracy [33] by carrying out OCR on original and rectified document images. Furthermore, experiments have been carried out using a semi-automatic evaluation methodology proposed in [25]. The experimental results from both procedures are presented in the sequel.

A. OCR Evaluation

The use of OCR as a means for indirect evaluation is widely used in the evaluation of rectification techniques [14], [16]–[20], [22], [24]. Character accuracy is defined as the ratio of the number of correct characters (number of characters in the correct document transcription minus the number of errors) over the total number of characters in the correct document transcription

$$Character\ Accuracy = \frac{(\#characters - \#errors)}{\#characters}. \quad (14)$$

In order to define the errors we count the minimum number of edit operations (insertion, deletion or substitution) that are required to correct the text generated by the OCR system (string edit distance). Furthermore, almost all commercial OCR systems when they have low confidence in their decision, they mark the character as "suspect". Consequently, in an OCR system processes, the better the document image quality is the higher confidence it has, since less suspect characters are produced.

We used a dataset of 100 distorted document images at 200 dpi, which contains both modern and historical printed

TABLE I
AVERAGE CHARACTER ACCURACY ON 100 DOCUMENT IMAGES

Rectification Technique	#characters	#errors	#suspect characters	Character Accuracy
Without Rectification	170726	74191	2739	56,54%
Gatos et al. [10]	170726	31553	3896	81,51%
Stamatopoulos et al. [11]	170726	24637	3467	85,56%
BookRestorer [34]	170726	16170	2627	90,52%
Proposed method	170726	**10549**	**879**	**93,82%**

document images in English and German language. The document images contain different font sizes and suffer from several distortions. The proposed methodology takes approximately 12 seconds on average for document rectification. For comparison purposes, we applied at the same dataset our previous works [10] and [11] as well as the commercial package BookRestorer [34]. OCR testing is performed using ABBYY FineReader Engine 8.1 [35]. Both the distorted document images and the rectified documents are fed into OCR Engine for text recognition. Table I illustrates the average character accuracy as well as the total suspect characters which had been marked by the OCR Engine.

As the recognition rates indicate, the rectified document images using the proposed methodology lead to higher performance in terms of OCR than the original document images. After rectification, the character accuracy is improved by 37% which demonstrates the effectiveness of the proposed rectification method. Firstly, as we can observe, the current proposed methodology outperforms our previous works [10] and [11] which directly ensures that the difference between our older works and the currently proposed is significant. Also, the proposed methodology outperforms the commercial package BookRestorer. Furthermore, as it is shown in Table I, the less suspect characters are generated by the proposed methodology. It means that the rectified document images produced by the proposed methodology have better quality and the confidence of the OCR Engine is increased. Furthermore, the small number of suspect characters indicates that if we use more document images the proposed methodology has a higher probability to produce better OCR results than the other methods which produce much more suspect characters. A representative result is shown in Fig. 12. The proposed methodology corrects all distortions, not only the skew but also the vertical alignment of the original document image. As shown in Fig. 12, the proposed method can also deal with non-text content. Nevertheless, it can be observed that there exist some errors due to erroneous word baseline estimation such as in the caption of the figure. However, these types of errors may not influence the OCR procedure. Our previous work [11] causes much more errors due to erroneous word baseline estimation and also many words are distorted because it does not split these words according to their baselines (see Section III-D). Furthermore, our previous work [10] merges some text lines mainly due to segmentation errors and it does not correct the vertical alignment of the document. Finally, BookRestorer cannot handle all distortions of the original document and several text lines are not straight.

Word detection stage is a crucial stage of the proposed rectification methodology (see Section III-C). For this reason, to

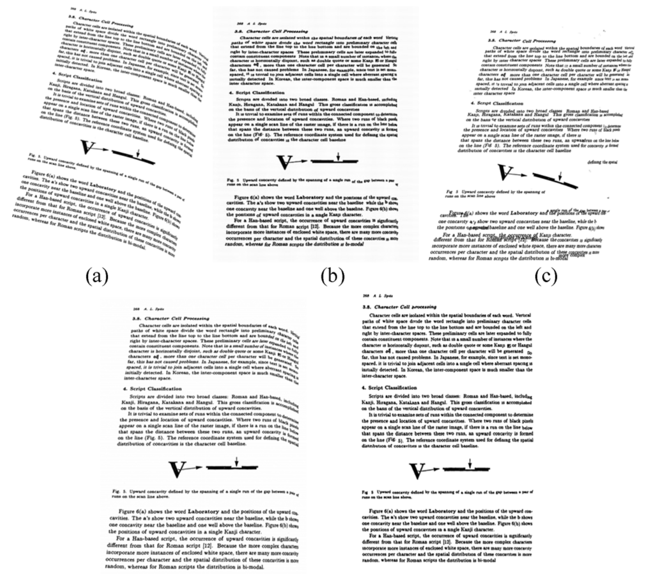


Fig. 12. Recovery of a distorted document images; (a) original document image; rectified document images using (b) work presented in [11]; (c) work presented in [10]; (d) commercial package BookRestorer [34]; (e) proposed methodology.

TABLE II
AVERAGE WORD ACCURACY ON 100 DOCUMENT IMAGES

Rectification Technique	#words	#misrecognized word	Word Accuracy
Without Rectification	27012	14916	44,78%
Gatos et al. [10]	27012	10072	62,71%
Stamatopoulos et al. [11]	27012	9169	66,06%
BookRestorer [34]	27012	5714	78,85%
Proposed method	27012	**4303**	**84,07%**

verify the validity and effectiveness of the proposed methodology, we carried out OCR testing on original and rectified document images using also the word accuracy measure. Word accuracy is defined as the ratio of the number of correct words (number of words in the correct document transcription minus number of misrecognized words) to the total number of word in the correct document transcription

$$\text{Word Accuracy} = \frac{(\#\,\text{words} - \#\,\text{misrecognized\_words})}{\#\,\text{words}}. \tag{15}$$

A word in the result is considered correct if all of its characters are correct. Table II illustrates the average word accuracy results. As in the previous experiment that takes into account the character accuracy, the proposed rectification method outperforms all the other methods with an improvement of 39%. Moreover, the overall comparative ranking is the same.

In order to examine how well the word detection stage performs, it is replaced with two state of the art techniques, the word detection used in FineReader [35] and the one used in OCROpus [36] and all the above evaluation process is repeated. Table III illustrates the overall evaluation results of the proposed rectification method using each time a different word detection technique. As the recognition rates show, the character and word accuracy of the rectified document images which produced using

TABLE III
AVERAGE OCR RATES PRODUCED BY THE PROPOSED RECTIFICATION METHOD USING DIFFERENT WORD DETECTION TECHNIQUES

Word Detection Technique	#characters	#errors	#suspect characters	Character Accuracy	#words	#misrecognized word	Word Accuracy
OCROpus	170726	13279	2109	92,22%	27012	5442	79,85%
FineReader	170726	49018	2162	71,28%	27012	10886	59,70%
Proposed method	170726	**10549**	**879**	**93,82%**	27012	**4303**	**84,07%**

TABLE IV
COMPARATIVE RESULTS USING THE SEMI-AUTOMATIC EVALUATION METHODOLOGY

Rectification Technique	*DW*
Gatos et al. [10]	80,55%
Stamatopoulos et al. [11]	82,90%
BookRestorer [34]	84,80%
Proposed method	**91,71%**

the proposed word detection technique are higher than using either the FineReader or OCROpus word detection techniques. Consequently, the proposed word detection technique is stable, efficient and demonstrates the robustness of the proposed rectification methodology.

B. Semi-Automatic Evaluation

The evaluation methodology proposed in [25] avoids the dependence on an OCR engine or human interference. It is based on a point-to-point matching procedure using Scale Invariant Feature Transform (SIFT) [37] as well as the use of cubic polynomial curves for the calculation of a comprehensive measure which reflects the entire performance of a rectification technique in a concise quantitative manner. First, the user manually mark specific points on the distorted document image which correspond to N appropriate text lines of the document with representative deformation. Then, using SIFT transform, the marked points of the distorted document image are matched to the corresponding points of rectified document image. Finally, the cubic polynomial curves which fit to these points are estimated and are taken into account in the evaluation measure DW

$$DW = \frac{\sum_{j=1}^{N} DW_j}{N} \times 100\% \tag{16}$$

where DW_j is the measure which reflects the performance of the rectification technique with respect to the jth selected text line. DW_j equals to one when the jth selected text line in the rectified document image is a horizontal straight text line that is the expected optimal result. It shows that the rectification technique produces the best result. On the other hand, DW_j equals to zero when the rectified document image is either equal or worse than the original image. Therefore, DW ranges in the interval $[0, \ldots, 100]$ and the higher the value of DW, the better is the performance of the rectification technique.

In our experiments, we used a subset of the document images which were used in OCR evaluation consisting of 30 document images. First, we manually marked six text lines ($N = 6$) with representative deformation instances at each document image and then we extracted the DW measure for all rectification methods. Table IV illustrates the average DW measure of all rectification methods. It is worth mentioning that the overall comparative ranking is the same with the one which is produced with the experiment that takes into account OCR performance. The proposed rectification method outperforms all the other methods.

V. CONCLUSION

A goal-oriented coarse-to-fine rectification methodology has been proposed in order to remove undesirable distortions from document images without auxiliary hardware or information. Experimental results on several distorted document images show that the proposed method produces rectified images that give a significant boost in OCR performance. The proposed methodology increases the confidence of the OCR Engine as it further reduces the suspect characters produced by the OCR Engine. Furthermore, experiments have been carried out using a new semi-automatic evaluation methodology. The evaluation results are in line with the OCR results demonstrating the effectiveness of the proposed rectification methodology. Finally, experiments have been carried out to identify the contribution of the proposed word detection method in the performance of rectification process against other state of the art word segmentation techniques. Our future research will focus on the extension of the proposed methodology in order to handle document images in which the text content is not justified and also contain cursive handwritten text.

APPENDIX A
MODELLING THE CURVED SURFACE PROJECTION ON THE PLANE

The detailed procedure in order to estimate the straight lines AD and BC which correspond to the left and right text boundaries as well as the curved lines AB and DC which correspond to the top and bottom text lines (see Fig. 5) is as follows:

Left and Right Straight Line Segments Estimation:

Step 1) Detect the start and end points of each text line: $((x_{s_i}, y_{s_i}), (x_{e_i}, y_{e_i}))$, $0 < i < NL$ where NL denote the number of lines.

Step 2) Calculate the length of each text line: $L_i = \sqrt{(x_{e_i} - x_{s_i})^2 + (y_{e_i} - y_{s_i})^2}$.

Step 3) Calculate the average length AL of text lines and every text line i is excluded if $L_i \leq 0.8{*}AL$.

Step 4) Using the text lines that remain after Step 3, a Least Squares Estimation (LSE) method is used to get the straight lines AD and BC that fit into set of points which denote the start and end points of each text line, respectively.

Step 5) For each straight line AD, BC we calculate the average Euclidean distance between the line and all points (start and end points of each text line).

If the average distance is greater than AH (dominant character height) we exclude the point with the maximum distance and repeat Steps 4 and 5 to recalculate the straight line till the abovementioned condition is satisfied.

Top and Bottom Curved Line Segments Estimation:

Step 1) Select the top and bottom text lines which take part in the calculation of the straight lines.

Step 2) Detect all the upper points of the top text line and the bottom points of the bottom text line.

Step 3) A Least Squares Estimation method is used to find the coefficients of 3rd degree polynomial curves that fit all top and bottom points detected in Step 2.

APPENDIX B
TRANSFORMATION MODEL

The distinct steps of the transformation are as follows:

Step 1) Calculate the width W and the height H of the rectangular area as follows:

$$W = \min(|\overset{\frown}{AB}|, |\overset{\frown}{DC}|) \ \& \ H = \min(|AD|, |BC|) \quad (17)$$

Step 2) Define the rectangular area $A'B'C'D'$: In order to locate the corner points of the rectangular area, one of them coincides with one of the dominant corner points of the projection of the curved surface on the plane $(A'(x_1', y_1') \equiv A(x_1, y_1))$ and the rest of them are calculated by taking into account the width W and the height H of the rectangular area (see Fig. 6) as follows:

$$(x'_2, y'_2) = (x'_1 + W, y'_1), \ (x'_3, y'_3) = (x'_1 + W, y'_1 + H) \\ \& \ (x'_4, y'_4) = (x'_1, y'_1 + H). \quad (18)$$

Step 3) Create a correspondence between the points of curved line segments AB and DC expressed by a function $\Im$ defined as follows:

$$G(x_b, y_b) = \Im(E(x_u, y_u)), \ \text{if} \ \frac{|\overset{\frown}{AE}|}{|\overset{\frown}{AB}|} = \frac{|\overset{\frown}{DG}|}{|\overset{\frown}{DC}|} \quad (19)$$

where $E(x_u, y_u)$ represent a point on the top curved line segment AB and $G(x_b, y_b)$ represent a point on the bottom curved line segment DC (see Fig. 6).

Step 4) Let $O(x, y)$ represent a point in the projection of the curved surface. Our goal is to calculate the new position $O'(x', y')$ of $O(x, y)$ in the rectangular area (see Fig. 6). Firstly, we define the straight line EG which satisfies the following criteria:

Criterion 1: Intersects curved lines AB and DC at the points $E(x_u, y_u)$ and $G(x_b, y_b)$, respectively.

Criterion 2: $G(x_b, y_b) = \Im(E(x_u, y_u))$.

Criterion 3: $O(x, y) \in \overline{EG}$

Then, we calculate the position of $O'(x', y')$ as follows:

$$x' = x_1' + |A'Z| \quad \& \quad y' = y_1' + |A'H| \quad (20)$$

where H is point $H(x_1', y')$ which belongs to the left side of the rectangular area, and Z is point $Z(x', y_1')$ which belongs to the top side of the rectangular area. Finally, $|A'Z|$ and $|A'H|$ are calculated using the following equations:

$$\frac{|\overset{\frown}{AB}|}{|\overset{\frown}{AE}|} = \frac{W}{|A'Z|} \Rightarrow |A'Z| = \frac{W}{|\overset{\frown}{AB}|}|\overset{\frown}{AE}| \ \& \\ \frac{|EG|}{|EO|} = \frac{H}{|A'H|} \Rightarrow |A'H| = \frac{H}{|EG|}|EO|. \quad (21)$$

Consequently, we repeat Step 4 for all points which are inside the projection area borders.

Step 5) All points which are not included at the projection area of the curved surface inherit the transformation of the nearest point.

REFERENCES

[1] J. Liang, D. Doermann, and H. Li, "Camera-based analysis of text and documents: A survey," *Int. J.Document Analysis and Recognition*, vol. 7, no. 2-3, pp. 84–104, 2005.

[2] L. Zhang, Y. Zhang, and C. L. Tan, "An improved physically-based method for geometric restoration of distorted document images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 4, pp. 728–734, Apr. 2008.

[3] A. Ulges, C. H. Lampert, and T. Breuel, "Document capture using stereo vision," in *Proc. ACM Symp. Document Eng.*, Milwaukee, WI, 2004, pp. 198–200.

[4] A. Yamashita, A. Kawarago, T. Kaneko, and K. T. Miura, "Shape reconstruction and image restoration for non-flat surfaces of document with a stereo vision system," in *Proc. 17th Int. Conf. Pattern Recognit.*, Cambridge, U.K., 2004, pp. 482–485.

[5] M. S. Brown and W. B. Seales, "Image restoration of arbitrarily warped documents," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 10, pp. 1295–1306, Oct. 2004.

[6] H. Cao, X. Ding, and C. Liu, "Rectifying the bound document image captured by the camera: A model based approach," in *Proc. 7th Int. Conf. Document Anal. Recognition*, 2003, pp. 71–75.

[7] J. Liang, D. DeMenthon, and D. Doermann, "Geometric rectification of camera-captured document images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 4, pp. 591–605, Apr. 2008.

[8] C. L. Tan, L. Zhang, Z. Zhang, and T. Xia, "Restoring warped document images through 3-D shape modeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 2, pp. 195–208, Feb. 2006.

[9] L. Zhang, A. M. Yip, M. S. Brown, and C. L. Tan, "A unified framework for document restoration using inpainting and shape-from-shading," *Pattern Recognit. J.*, vol. 42, no. 11, pp. 2961–2978, 2009.

[10] B. Gatos, I. Pratikakis, and K. Ntirogiannis, "Segmentation based recovery of arbitrarily warped document images," in *Proc. 9th Int. Conf. Document Anal. Recognit.*, Curitiba, Brazil, 2007, pp. 989–993.

[11] N. Stamatopoulos, B. Gatos, I. Pratikakis, and S. J. Perantonis, "A two-step dewarping of camera document images," in *Proc. 8th Int. Workshop Document Analysis Syst.*, Nara, Japan, 2008, pp. 209–216.

[12] O. Lavialle, X. Molines, F. Angella, and P. Baylou, "Active contours network to straighten distorted text lines," in *Proc. Int. Conf. Image Process.*, Thessaloniki, Greece, 2001, pp. 748–751.

[13] C. Wu and G. Agam, "Document image De-warping for Text/Graphics recognition," in *Joint IAPR Int. Workshop Structural, Syntactic Statistical Pattern Recognit.*, Windsor, Canada, 2002, pp. 348–357.

[14] L. Zhang and C. L. Tan, "Warped image restoration with applications to digital libraries," in *Proc. 8th Int. Conf. Document Anal. Recognit.*, Seoul, Korea, 2005, pp. 192–196.

[15] H. Ezaki, S. Uchida, A. Asano, and H. Sakoe, "Dewarping of document image by global optimization," in *Proc. 8th Int. Conf. Document Analysis Recognit.*, Seoul, Korea, 2005, pp. 302–306.

[16] L. Mischke and W. Luther, "Document image de-warping based on detection of distorted text lines," in *Proc. Int. Conf. Image Anal. Process.*, Cagliari, Italy, 2005, pp. 1068–1075.

[17] A. Ulges, C. H. Lampert, and T. M. Breuel, "Document image dewarping using robust estimation of curled text lines," in *Proc. 8th Int. Conf. Document Anal. Recognit.*, Seoul, Korea, 2005, pp. 1001–1005.

[18] S. J. Lu, B. M. Chen, and C. C. Ko, "A partition approach for the restoration of camera images of planar and curled document," *Image Vis. Comput.*, vol. 24, no. 8, pp. 837–848, 2006.

[19] D. C. Schneider, M. Block, and R. Rojas, "Robust document warping with interpolated vector fields," in *Proc. 9th Int. Conf. Document Anal. Recognit.*, Curitiba, Brazil, 2007, pp. 113–117.

[20] S. S. Bukhari, F. Shafait, and T. M. Breuel, "Dewarping of document images using coupled-snakes," in *Proc. Int. Workshop Camera-Based Document Anal. Recognit.*, Barcelona, Spain, 2009, pp. 34–41.

[21] B. Fu, M. Wu, R. Li, W. Li, Z. Xu, and C. Yang, "A model-based book dewarping method using text line detection," in *Proc. Int. Workshop Camera-Based Document Anal. Recognit.*, Curitiba, Brazil, 2007, pp. 63–70.

[22] Y. Zhang, C. Liu, X. Ding, and Y. Zou, "Arbitrary warped document image restoration based on segmentation and Thin-Plate Splines," in *Proc. 19th Int. Conf. Pattern Recognit.*, 2008, pp. 1–4.

[23] M. S. Brown and Y. C. Tsoi, "Geometric and shading correction for images of printed materials using boundary," *IEEE Trans. Image Process.*, vol. 15, no. 6, pp. 1544–1554, Jun. 2006.

[24] A. Masalovitch and L. Mestetskiy, "Usage of continuous skeletal image representation for document images de-warping," in *Proc. Int. Workshop Camera-Based Document Anal. Recognit.*, Curitiba, Brazil, 2007, pp. 45–53.

[25] N. Stamatopoulos, B. Gatos, and I. Pratikakis, "A methodology for document image dewarping techniques performance evaluation," in *Proc. 10th Int. Conf. Document Anal. Recognit.*, Barcelona, Spain, 2009, pp. 956–960.

[26] T. M. Breuel, "Two geometric algorithms for layout analysis," in *Proc. Int. Workshop Document Anal. Syst.*, Princeton, NJ, 2002, pp. 188–199.

[27] B. Gatos, I. Pratikakis, and S. J. Perantonis, "Adaptive degraded document image binarization," *Pattern Recognit. J.*, vol. 39, no. 3, pp. 317–327, 2006.

[28] N. Stamatopoulos, B. Gatos, and A. Kesidis, "Automatic borders detection of camera document images," in *Proc. Int. Workshop Camera-Based Document Anal. Recognit.*, Curitiba, Brazil, 2007, pp. 71–78.

[29] F. M. Wahl, K. Y. Wong, and R. G. Casey, "Block segmentation and text extraction in mixed text/image documents," *Comput. Graph. Image Process.*, vol. 20, no. 4, pp. 375–390, 1982.

[30] G. Farin*, Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*, 4th ed. New York: Academic, 1996.

[31] S. Coons, "Surfaces for computer aided design," Tech. Rep., Mass. Inst. Technol., Cambridge, 1968.

[32] U. V. Marti and H. Bunke, "Using a statistical language model to improve the performance of an HMM-based cursive handwriting recognition system," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 15, no. 1, pp. 65–90, 2001.

[33] J. Kanai, T. A. Nartker, S. Rice, and G. Nagy, "Performance metrics for document understanding systems," in *Proc. 2nd Int. Conf. Document Anal. Recognit.*, Japan, 1993, pp. 424–427.

[34] BookRestorer, [Online]. Available: http://www.i2s-bookscanner.com/

[35] ABBYY FineReader OCR, [Online]. Available: http://finereader.abbyy.com/

[36] The OCROpus Open Source Document Analysis System, [Online]. Available: http://code.google.com/p/ocropus/

[37] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

Nikolaos Stamatopoulos was born in Athens, Grecce in 1984. He graduated in 2006 from the Department of Informatics and Telecommunications of National and Kapodistrian University of Athens. From November 2006 he is a Ph.D. candidate at the same university. Currently, he is working at the Institute of Informatics and Telecommunications in the National Center for Scientific Research "Demokritos", Athens, Greece. His main research interests are in image processing, document image analysis and processing of historical documents.

Basilis Gatos received his Electrical Engineering Diploma in 1992 and his Ph.D. degree in 1998, both from the Electrical and Computer Engineering Department of Democritus University of Thrace, Xanthi, Greece. He is currently working as a researcher at the Institute of Informatics and Telecommunications of the National Center for Scientific Research "Demokritos", Athens, Greece. His main research interests are in image processing, pattern recognition, document image analysis, OCR, processing and recognition of historical documents.

Ioannis Pratikakis (M'88) is an Assistant Professor at the Department of Electrical and Computer Engineering, Democritus University of Thrace, Xanthi, Greece. He received the Ph.D. degree in 3-D Image analysis from the Electronics and Informatics engineering department at Vrije Universiteit Brussel, Belgium, in January 1999. From March 1999 to March 2000, he was at IRISA/ViSTA group, Rennes, France as an INRIA postdoctoral fellow. Since 2003, he is working as Adjunct Researcher at the Institute of Informatics and Telecommunications in the National Centre for Scientific Research "Demokritos", Athens, Greece. His research interests include multidimensional document image analysis, 3-D computer vision, graphics and multimedia search and retrieval with a particular focus on visual content. He is a member of the IEEE Signal Processing Society and the European Association for Computer Graphics (Eurographics).

Stavros J. Perantonis is the holder of a BS degree in Physics from the Department of Physics, University of Athens, an M.Sc. degree in Computer Science from the Department of Computer Science, University of Liverpool and a D.Phil. degree in Computational Physics from the Department of Physics, University of Oxford. Since 1992 he has been with the Institute of Informatics and Telecommunications, NCSR "Demokritos", where he currently holds the position of Senior Researcher and Head of the Computational Intelligence Laboratory. His main research interests are in image processing and document image analysis, OCR and pattern recognition.