H-DocPro: A Document Image Processing Platform for Historical Documents

Basilis Gatos
Computational Intelligence Laboratory
Institute of Informatics and Telecommunications, NCSR "Demokritos"
GR-153 10 Agia Paraskevi, Athens, Greece
bgat@iit.demokritos.gr

Nikolaos Stamatopoulos
Computational Intelligence Laboratory
Institute of Informatics and Telecommunications, NCSR "Demokritos"
GR-153 10 Agia Paraskevi, Athens, Greece
nstam@iit.demokritos.gr

Georgios Louloudis
Computational Intelligence Laboratory
Institute of Informatics and Telecommunications, NCSR "Demokritos"
GR-153 10 Agia Paraskevi, Athens, Greece
louloud@iit.demokritos.gr

Stavros Perantonis
Computational Intelligence Laboratory
Institute of Informatics and Telecommunications, NCSR "Demokritos"
GR-153 10 Agia Paraskevi, Athens, Greece
sper@iit.demokritos.gr

ABSTRACT

In this paper, we introduce the H-DocPro platform which is a publicly available document image processing platform for historical documents. H-DocPro is a result of our recent and on-going research on historical document image processing and has been developed in order to monitor the successive application of several new or state-of-the-art document image processing methods. It is an open architecture software platform that permits several document image processing modules and methods (e.g. binarization, image enhancement, page split) to be utilized in an easy to define processing workflow. We provide detailed information on how to use H-DocPro, the available modules and methods as well as the way one can add his own components exploiting the open architecture form of the platform. Representative examples and experimental results using large sets of historical document images demonstrate the efficiency of H-DocPro methods.

Categories and Subject Descriptors

I.4.6 [Image Processing]: Segmentation; I.4.9 [Image Processing]: Applications; I.5.4 [Pattern Recognition]: Applications---Text Processing; I.7.5 [Document and Text Processing]: Document Capture---Document Analysis.

General Terms

Algorithms, Performance, Design, Experimentation.

DATeCH 2014, May 19-20 2014, Madrid, Spain
ACM 978-1-4503-2588-2/14/05.
http://dx.doi.org/10.1145/2595188.2595203

Keywords

Historical Document Image Processing; Software Platform; Workflow; Open Architecture Software

1. INTRODUCTION

Historical document image processing is a challenging and important topic of research since it permits the efficient exploitation of valuable historical collections. This is also proved by the fact that it is still an open research field that attracts the attention of several publications in the literature [1-3] as well as research projects [4-6]. During our recent involvement in such projects ([4],[6]), we came to the need of monitoring the successive application of several new or state-of-the-art methods for historical document image processing. Existing tools, such as the HistDoc platform [7] and the SCRIBO module of the Olena platform [8] either do not offer the capability of adding new components or need the experience of an expert to use it. Other tools, such as OCRopus open source system [9] and Taverna Workflow Management system [10], either do not offer an "easy to use" and "document image processing oriented" interface or need lot of programming efforts in order to handle modules monitoring. To this end, we developed and made publicly available the H-DocPro platform [11] which is an open architecture document image processing platform for historical documents (see Fig. 1). All necessary files in order to install H-DocPro for Windows O/S can be found at [11] (tested to work well with Windows XP, 7 and 8, 32 and 64-bit versions). An email has to be sent to the development team of H-DocPro in order to get a username/password for the application.

H-DocPro permits several document image processing modules and methods to be utilized in an easy to define processing workflow. Currently, there are several H-DocPro modules available for binarization, border removal, page split, image enhancement and dewarping. The strength of H-DocPro is that one can easily wrap and add his own document image processing components without making any changes to the main platform application.

In the following, information on how to use H-DocPro as well as on how to install new methods is provided in Sections 2 and 3. The available H-DocPro document image processing methods are detailed in Section 4, while in Section 5 experimental results using the H-DocPro methods and large sets of historical machine-printed documents are presented. Finally, conclusions are drawn in Section 6.

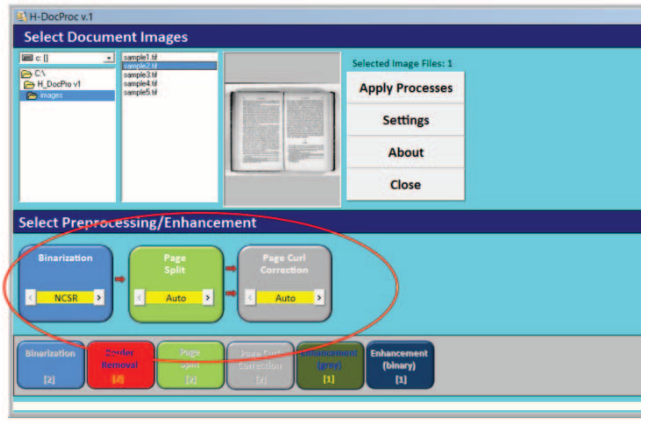


Figure 1. The H-DocPro platform. The modules added to the workflow pane are included in the red ellipses.

2. USING THE H-DOCPRO PLATFORM

To start working and explore the functionality of H-DocPro, 7 simple steps are necessary:

Step 1: Select the directory which contains the images for processing (input folder) or copy these images to directory [Install Dir]/Images.

Step 2: Select the directory for saving the resulting images after the application of a workflow (output folder) using the "Settings" button (default save directory: [Install Dir]/Results).

Step 3: Select one or more document images for processing.

Step 4: Define a processing workflow. In order to define a processing workflow, click on the corresponding modules and add them on the workflow pane (see Fig. 1).

Step 5: Select the method for every processing module by pressing "<" or ">" on every module at the workflow pane. Right click on the module at the workflow pane and (a) deselect "Do not recalculate if result exists" if you want to recalculate an existing result, (b) select "Settings" to change the parameters of the selected method (available for specific modules).

Step 6: The execution of the workflow starts after pressing the "Apply Processes" button.

Step 7: The user can view the final results for each selected image of step 3 using the preview window. Additionally, by right clicking on any module at the workflow pane and selecting "View Result", the user is able to see the intermediate result synchronized with the original image.

3. ADDING NEW COMPONENTS

Due to the open architecture of the H-DocPro platform, except for the initial components which are provided, a user can also add his own components. Two steps are necessary for the addition of new components: (i) creating new modules (optional step in the case that the module already exists) and (ii) creating a new method bounded with a module. A more detailed description of the two steps is provided in the following subsections.

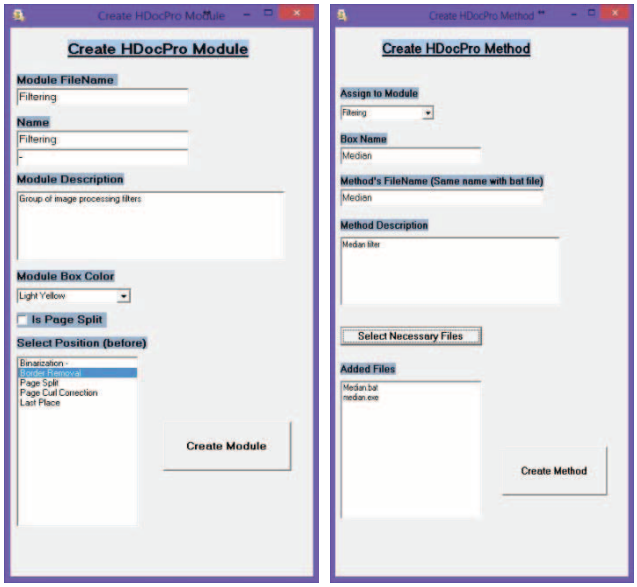
3.1 Creating New Modules

Every new method included in the H-DocPro platform needs to be assigned to the appropriate category of methods which is called module (e.g. binarization, border removal, page curl correction). If the module already exists (e.g. the user wants to add a new binarization method to the platform), the module creation step should be omitted. Otherwise, a new module should be created ("Settings"/"Create Module"). A screenshot of the module creation form is presented in Fig.2a. In this form, the user should define general information associated with the module including: (a) the filename of the module, (b) the name displayed in the platform (2 lines), (c) a general description for the module, (d) the color of the box module in the platform, (e) an indication that the module has page split properties (necessary information due to the creation of two images instead of one) and (f) the order of the new module in the platform pane.

3.2 Defining New Methods

In order to define a new method the user should have already created an executable of the method in the form of win32 console application which will take 2 arguments namely the input and output image. A batch file ([method].bat) should also exist defining the call of the executable. The definition of a new method for the H-DocPro platform involves the following information (see Fig.2b): (a) the module to which it should be attached, (b) the name of the method in the box, (c) the method's filename (note that this name should be the same with the batch file's name), (d) a description of the method and (e) all files that are necessary for the method to execute (e.g. dlls, libs) including both the console application and the batch file.

After the successful creation of a module and/or a method, H-DocPro needs to be restarted in order for the changes to be applied.



(a) (b)

Figure 2. (a) Module creation form, (b) Method definition form.

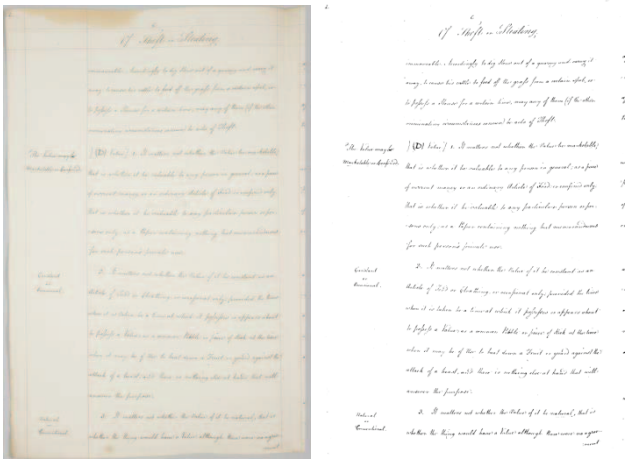
4. DOCUMENT IMAGE PROCESSING METHODS

In this section, all available H-DocPro document image processing methods for binarization, border removal, page split, image enhancement and dewarping are detailed.

4.1 Binarization

Document image binarization is an important pre-processing step of the document image processing and analysis pipeline and is defined as the process of segmenting the document image into text and background by removing any existing degradations. Since the majority of all methods of the subsequent document image processing and recognition tasks (e.g. border removal, page split, layout analysis, character recognition) use as input binary images, it is important to have a binarization result of enhanced quality that preserves text areas.

The "Binarization" module includes two different binarization methods, namely the "NCSR" method which is based on [12] (see Fig. 3) and the "FR8.1" method which implements a call to the FineReader Engine v. 8.1 [13]. The "NCSR" method [12] is an adaptive approach for the binarization and enhancement of historical degraded documents that suffer from shadows, non-uniform illumination, low contrast, large signal-dependent noise, smear and strain. It uses a pre-processing procedure using a low-pass Wiener filter, a rough estimation of foreground regions, a background surface calculation by interpolating neighboring background intensities, a thresholding by combining the calculated background surface with the original image and, finally, a post-processing step in order to improve the quality of text regions and preserve stroke connectivity.

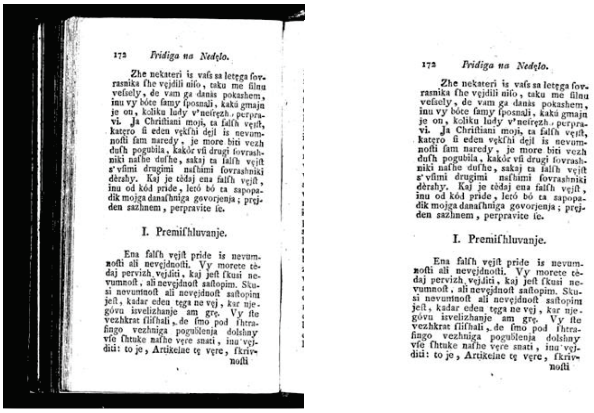


(a) (b)

Figure 3. Example of "NCSR" binarization method: (a) original document image; (b) binarization result.

4.2 Border Removal

Document images usually contain two main types of marginal noise: (i) non-textual noise resulting from the page surrounding and the binarization process and (ii) textual noise from neighboring pages (see Fig. 4a). These types of marginal noise affect the performance of subsequent processing. The "Border Removal" method detects and removes both types of marginal noise. It is based on projection profiles combined with a connected component labeling process. Signal cross-correlation is also used in order to verify the detected textual noise [14]. A representative example of border removal is shown in Fig. 4b. Apart from the fully automatic procedure (method "Auto"), a semi-automatic procedure (method "Auto_Edit") is also available. The user, after a visual inspection of the result, is able to refine it by changing the marked area.



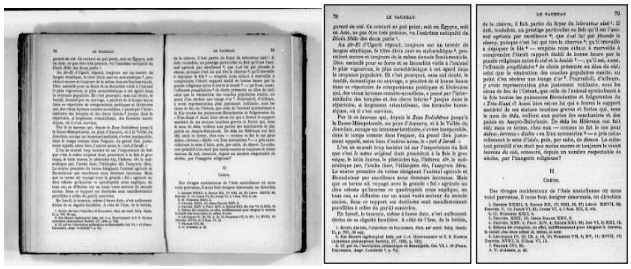
(a) (b)

Figure 4. Example of the "Border Removal" method: (a) original document image; (b) document image after applying border removal.

4.3 Page Split

Scanning two pages at the same time is a very common practice as it helps to accelerate the scanning process. However, it may affect the performance of subsequent processing modules since the majority of approaches is able to process single page document images. Furthermore, another drawback of scanning two pages at the same time is the appearance of noisy black borders around text areas as well as of noisy black stripes between two pages (see Fig.5a).

The "Page Split" method, which is able to process binary, grayscale or color images, detects the optimal page frames of double page document images, splits the image into two pages and removes noisy borders (see Fig. 5). The page split method is based on the vertical and horizontal white run projections which have been proved efficient for detecting zones of text [15]. Apart from the fully automatic page split procedure (method "Auto"), the user is also able to refine the page split result by adjusting the page frames (method "Auto_Edit").



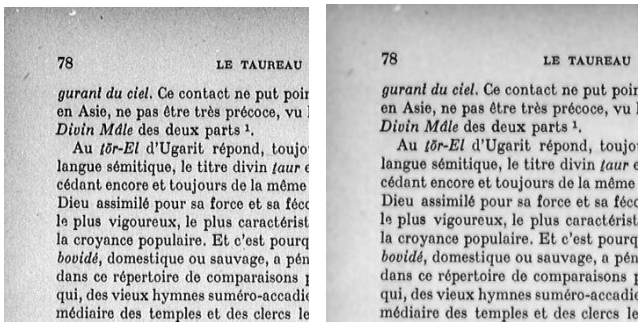
(a) (b) (c)

Figure 5. Example of "Page Split" method: (a) original double page document image; (b)-(c) output document images after applying page split.

4.4 Image Enhancement

Historical document images usually suffer from several defects mainly due to natural ageing, environmental conditions, usage as well as poor storage conditions of the original documents. Furthermore, binarization may result to unwanted noisy regions. To this end, image enhancement is necessary to improve the quality of historical document images. For H-DocPro, we have implemented two image enhancement modules, one for gray scale ("Enhancement (gray)") and one for binary ("Enhancement (binary)") images.

For the enhancement of gray scale images, we selected the application of the Wiener filter ([16]) (method "Wiener" - see Fig. 6). Wiener filter is commonly used in filtering theory for image restoration. It can be applied to degraded and poor quality grayscale documents in order to eliminate noisy areas, smooth the background texture as well as enhance the contrast between background and text areas. For the enhancement of binary images, we implemented a despeckle filter (method "Despeckle"). Despeckling is the operation of removing unwanted small components of the binary image. The window size of the Wiener filter as well as the maximum size of the speckle noise component to be removed can be set by activating the settings of the corresponding module.



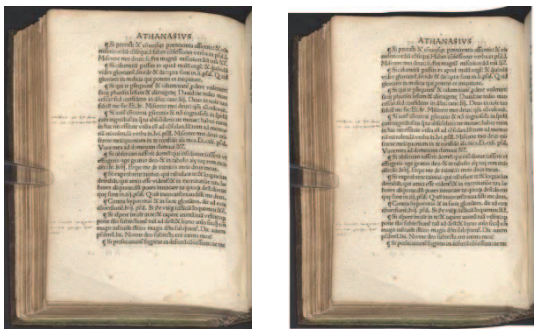
(a) (b)

Figure 6. Example of "Wiener" method: (a) original document image; (b) output document image after applying enhancement based on Wiener filter.

4.5 Dewarping

Document image acquisition by a flatbed scanner or a digital camera often results in several unavoidable image distortions due to the form of printed material (e.g. bounded volumes), the camera setup or environmental conditions (e.g. humidity that causes page shrinking). The "Dewarping" method removes undesirable distortions from single column document images (see Fig. 7). At a first step, a coarse dewarping is accomplished with the help of a transformation model which maps the projection of a curved surface to a 2D rectangular area. The curved surface 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. At a second step, fine dewarping is achieved based on word detection [17].

Apart from the fully automatic dewarping procedure (method "Auto"), the user is able to correct manually the position of the curved and straight lines, by dragging the corresponding black points, in order to improve the modeling of the curved surface projection that will consequently lead to a more successful coarse rectification result (method "Auto_Edit"). Once the curved surface projection has been defined, the user can test the final result by pressing the button "Dewarping" (see Fig. 8). Moreover, the number of points that used to model the two curved lines can be defined by the user from the Setting options. This may lead to a more precise modeling of the two curved lines.



(a) (b)

Figure 7. Example of applying the "Dewarping" method: (a) original document image; (b) document image after dewarping.

Figure 8. Screenshot of the dewarping semi-automatic method ("Auto_Edit"). The curved surface is delimited by the two straight lines (red and blue) and two curved lines (green and black).

5. EXPERIMENTAL RESULTS

Evaluation results for the main document image processing methods are presented in this section in order to prove their efficiency.

5.1 *Border Removal*

In order to record the efficiency of the border removal method, we manually marked the correct text region in the original image using a polygon mark-up tool for creating the ground truth. Performance evaluation is accomplished using a pixel based approach which counts the number of pixels at the correct as well as at the resulting text region area. Let G be the set of all pixels inside the correct text region in the ground truth, R the set of all pixels inside the resulting image and T(s) a function which counts the elements of set s. We calculate the Precision and Recall using the following equations:

$$Precision = \frac{T(G \cap R)}{T(R)}\ ,\ Recall = \frac{T(G \cap R)}{T(G)} \qquad (1)$$

A performance metric FM can be extracted by combining the values of precision and recall:

$$FM = \frac{2 * Precision * Recall}{Precision + Recall} \quad (2)$$

For the creation of the experimental set, 38718 historical document images were randomly selected from the IMPACT Dataset [4]. For comparison purposes, we also applied the state-of-the-art method [18] (D.X. Le) as well as the commercial products BookRestorer [19], WiseBook [20] and ScanFix [21]. Table 1 illustrates the average precision, recall and FM of all methods. As it can be observed, the H-DocPro border removal method [14] outperforms all other methods and achieves an *FM* of 98.93%.

Table 1. Border Removal Evaluation Results

Method	Precision (%)	Recall (%)	FM (%)
H-DocPro [14]	99.08	98.79	98.93
D.X Le [18]	98.30	96.63	97.30
BookRestorer [19]	96.47	97.06	96.76
WiseBook [20]	90.20	98.56	94.20
ScanFix [21]	91.17	97.46	94.21

5.2 *Page Split*

Concerning the page split method [15], we were based on the performance evaluation method which is described in Section 5.1. We used a set of 3467 double page document images of historical books from the IMPACT Dataset [4]. Table 2 illustrates the evaluation results of the H-DocPro page split method with an overall performance of 95.09% in terms of FM.

Table 2. Page Split Evaluation Results

Method	*Precision (%)*	*Recall (%)*	*FM (%)*
H-DocPro [15]	92.04	98.35	95.09

5.3 *Dewarping*

In order to measure the performance of the H-DocPro dewarping method we used the evaluation methodology presented in [22]. This methodology avoids the dependence on an OCR engine or human interference. It is based on a point-to-point matching procedure using the Scale Invariant Feature Transform (SIFT) as well as cubic polynomial curves for the calculation of a comprehensive measure which reflects the entire performance of a rectification technique in a concise quantitative manner. At a first step, the user manually marks specific points on the distorted document image which correspond to N appropriate text lines of the document with representative deformation. Using the SIFT transform, the marked points of the distorted document image are matched to the corresponding points of the rectified document image. Finally, the cubic polynomial curves which fit to the selected text lines are estimated and the dewarping evaluation measure (DM) is calculated based on the integral of each curve [22].

For our experiments, we used 420 randomly selected historical document images from the IMPACT Dataset [4] and compared with the commercial product BookRestorer [19]. First, we manually marked six text lines (N=6) with representative deformations at each document image and then we extracted the DM measure. The overall evaluation results are presented in Table 3. As it can be observed, the H-DocPro dewarping method [17] performs significantly better than BookRestorer (~7%).

Table 3. Dewarping Evaluation Results

Method	*DM (%)*
H-DocPro [17]	87.78
BookRestorer [19]	80.87

6. CONCLUSIONS

H-DocPro is a publicly available document image processing platform for historical documents developed in order to monitor the successive application of several new or state-of-the-art methods for historical document image processing. Several document image processing modules and methods have been already developed for H-DocPro (binarization, border removal, page split, image enhancement and dewarping modules) while new document image processing components can be easily wrapped and added without making any changes to the main platform application. In order to prove the efficiency of H-DocPro methods, we provide several experiments using large sets of historical machine-printed document images.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreements n° 215064 – project IMPACT and n° 600707 – project tranScriptorium.

7. REFERENCES

[1] M. Coustaty, R. Pareti, N. Vincent and J. M. Ogier, "Towards historical document indexing: extraction of drop cap letters", International Journal on Document Analysis and Recognition, 14(3), pp 243-254, 2011.

[2] K. Ntirogiannis, B. Gatos and I. Pratikakis, "A Performance Evaluation Methodology for Historical Document Image Binarization", IEEE Transactions on Image Processing, 22(2), pp. 595-609, 2013.

[3] H. Wei, G. Gaom "A keyword retrieval system for historical Mongolian document images", International Journal on Document Analysis and Recognition, Vol. 17, No.1, pp. 33-45, 2014.

[4] IMPACT project: http://www.impact-project.eu/

[5] Europeana Newspapers project: http://www.europeana-newspapers.eu/

[6] tranScriptorium project: http://transcriptorium.eu/

[7] R. Lins, G. Silva, and A. Formiga, "HistDoc v. 2.0: enhancing a platform to process historical documents". Workshop on Historical Document Imaging and Processing (HIP '11), pp. 169-176, 2011.

[8] G. Lazzara, R. Levillain, T. Géraud, Y. Jacquelet, J. Marquegnies, and A. Crepin-Leblond, "The SCRIBO Module of the Olena Platform: A Free Software Framework for Document Image Analysis", 11th International

Conference on Document Analysis and Recognition (ICDAR 2011), pp.252-258, 2011.

[9] OCRopus open source document analysis and OCR system: http://code.google.com/p/ocropus/

[10] Taverna Workflow Management system: http://www.taverna.org.uk/

[11] H-DocPro v.1, A Document Image Processing Platform for Historical Documents: http://www.iit.demokritos.gr/~bgat/H-DocPro/

[12] B. Gatos, I. Pratikakis and S. J. Perantonis, "Adaptive Degraded Document Image Binarization", Pattern Recognition, Vol. 39, pp. 317-327, 2006.

[13] ABBYY: http://www.abbyy.com/

[14] N. Stamatopoulos, B. Gatos and A. Kesidis, "Automatic Borders Detection of Camera Document Images", 2nd International Workshop on Camera-Based Document Analysis and Recognition, pp.71-78, 2007.

[15] N. Stamatopoulos, B. Gatos and T. Georgiou, "Page Frame Detection for Double Page Document Images", 9th International Workshop on Document Analysis Systems, pp. 401-408, 2010.

[16] A. Jain, Fundamentals of Digital Image Processing, Prentice-Hall, Englewood Cliffs, NJ, 1989.

[17] N. Stamatopoulos, B. Gatos, I. Pratikakis and S.J. Perantonis, "Goal-oriented Rectification of Camera-Based Document Images", IEEE Transactions on Image Processing, 20 (4), pp. 910-920, 2011.

[18] D.X. Le and G.R. Thoma, "Automated Borders Detection and Adaptive Segmentation for Binary Document Images". International Conference on Pattern Recognition, p. III: 737-741, 1996.

[19] Book Restorer (i2S): (http://www.i2s-bookscanner.com/

[20] WiseBook (CSoft: http://www.csoft.com/products/wisebook/

[21] ScanFix (accusoft pegasus): http://www.accusoft.com/scanfix.htm

[22] N. Stamatopoulos, B. Gatos and I. Pratikakis, "A Performance Evaluation Methodology for Document Image Dewarping Techniques", IET Image Processing, vol. 6, no. 6, pp. 738-745, 2012.