Improving Access to Text
IMPACT

Technical deliverable documentation

D-TR1 – IMAGE ENHANCEMENT TOOLKIT incl.
D-TR1.4(i) – Evaluation Report

Document history

Revisions

Version	Status	Author	Date	Changes
0.1	Draft	Vasily Panferov, Basilis Gatos, Stefan Pletschacher, Apostolos Antonacopoulos	31 May 2011	Created
0.2	Draft	“	20 July 2011	D-TR1.4c included
1.0	Final	“	07 December 2011	Review comments incorporated

Approvals

This document requires the following approvals:

Version	Date of approval	Name	Role in project	Signature
0.2	5 December 2011	Xavi Ivars	CB5 tool patron	OK
0.2	23 November 2011	Clemens Neudecker	Technical Project leader	OK
1.0	7 December 2011	Günter Mühlberger	SP leader TR	OK
1.0	7 December 2011	Hildelies Balk	General PM	OK

Distribution

This document was sent to:

Version	Date of sending	Name	Role in project
0.2	11 November 2011	Xavi Ivars, Clemens Neudecker,	Internal reviewers
1.0	7 December 2011	Günter Mühlberger, Hildelies Balk	SP leader TR, General PM
1.0	8 December 2011	Liina Munari	EC Project Officer

Contents:

IMPACT is supported by the European Community under the FP7 ICT Work Programme. The project is coordinated by the National Library of the Netherlands

IMPACT – Technical deliverable documentation

D-TR1 - Image enhancement toolkit

D-TR1.4(i) Evaluation report

This deliverable comprises three independent software packages which are aiming at specific aspects of image enhancement related to OCR. In the following, each tool is devoted a separate section including evaluation and final results.

1. Binarisation/Colour Reduction Toolkit

1.1. Partner

ABBYY

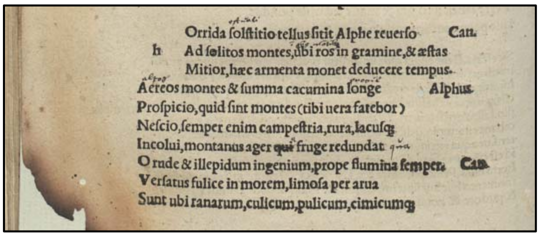
1.2. Deliverable

D-TR1(a). The toolkit is released on the basis of ABBYY FineReader Engine 10 for Windows.

1.3. Background

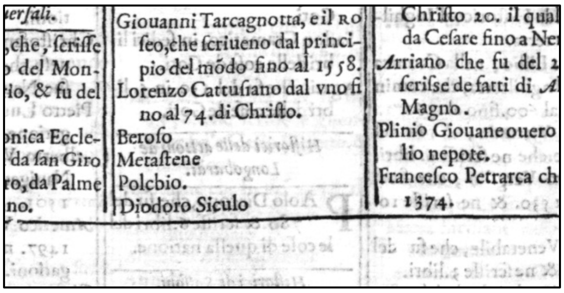
The main objective of the binarisation toolkit is to improve recognition quality of documents. Recognition quality can be affected very significantly by uneven page background, the presence of noise in images and other artefacts like bleed-through from the back page.

Some examples of distorted images:



Orrida ſolſtitio tellus ſitit Alphe reuerſo Can.
h Ad ſolitos montes, ubi ros in gramine, & æſtas
Mitior, hæc armenta monet deducere tempus.
Aereos montes & ſumma cacumina longe Alphus.
Proſpicio, quid ſint montes (tibi uera fatebor)
Neſcio, ſemper enim campeſtria, rura, lacuſq;
Incolui, montanus ager qui fruge redundat
O rude & illepidum ingenium, prope flumina ſemper. Can.
Verſatus fulice in morem, limoſa per arua
Sunt ubi ranarum, culicum, pulicum, cimicumq;

Figure 1: A page with uneven background

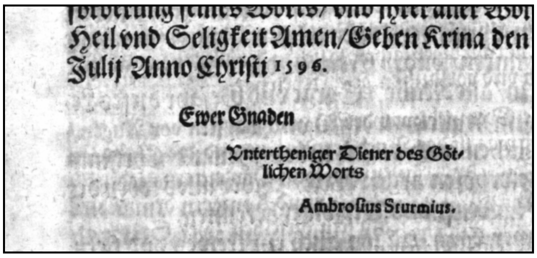


uerſali.
,che, ſcriſſe
del Mon-
io, & fu del
nica Eccle-
da ſan Giro
ro, da Palme
no.

Giouanni Tarcagnotta, e il RO
ſeo, che ſcriueno dal princi-
pio del mõdo fino al 1558.
Lorenzo Cattuſiano dal vno fi
no al 74. di Chriſto.
Beroſo
Metaſtene
Polcbio.
Diodoro Siculo

Chriſto 20. il qual
da Ceſare fino a Ne
Arriano che fu del 2
ſeriſse de fatti di A
Magno.
Plinio Giouane ouero
lio nepote.
Franceſco Petrarca ch
1374.

Figure 2: Bleed-through from the back page



ſorderung ſeines Worts/ vnd jhrer aller Wol
Heil vnd Seligkeit Amen/ Geben Krina den
Julij Anno Chriſti 1596.

Ewer Gnaden

Vntertheniger Diener des Göt-
lichen Worts

Ambroſius Sturmius.

Figure 3: General noise and bleed-through

IMPACT is supported by the European Community under the FP7 ICT Work Programme. The project is coordinated by the National Library of the Netherlands

To better deal with these issues, a new binarisation algorithm was developed. The previous version worked by collecting various statistics from different parts of an image. Using those statistics the original was converted to a bitonal image. The new version works with clusters of objects in the image. It tries to classify these objects and depending on their type utilises different techniques in their proximity. Here are some (not all) steps that the new version performs:

- The original image is smoothed internally. This allows ignoring fine noise like paper texture and unevenness of background color.
- Edges of the objects in the image are extracted using gradient information. With this technique it became possible to better binarise images with different brightness in different parts.
- Text-detection heuristics: Binarisation works differently on image parts which are assumed to represent text.

The binarisation algorithm comprises two stages. The first one is performed for better segmentation and image quality in general while the second one is executed during the recognition phase affecting the quality of the recognised text. It is only possible to visualise results of the first part since the second one is tightly integrated with the recognizer. The impact of the second stage can be evaluated indirectly by assessing the overall text recognition accuracy.

1.4. Outline of functionality

The Binarisation Toolkit was released as part of ABBYY FineReader Engine 10 SDK. This SDK is a tool that is already used for mass digitization tasks. It is thoroughly documented and contains many samples that demonstrate how to use it on different platforms and with different programming languages.

Here is a brief extract taken from the sample code that goes along with the toolkit:

```
// Add image to document
CBstr imageFilePath = L"SampleImages\\sample.tif";


CSafePtr<IFRDocument> frDoc;
CheckResult( FREngine->CreateFRDocument( &frDoc ) );
CheckResult( frDoc->AddImageFile( imageFilePath, 0, 0 ) );

// Recognize document internally using binarisation toolkit
CheckResult( frDoc->Process() );

// Get bitonal image
CSafePtr<IFRPages> pages;
CheckResult( frDoc->get_Pages( &pages ) );
long pageCount;
CheckResult( pages->get_Count( &pageCount ) );
for( int pageIndex = 0; pageIndex < pageCount; pageIndex++ ) {
      CSafePtr<IFRPage> page;
      CheckResult( pages->Item( pageIndex, &page ) );

      CSafePtr<IImageDocument> imageDoc;
      CheckResult( page->get_ImageDocument( &imageDoc ) );
      CSafePtr<IImage> bwImage;
```

```
        CheckResult( imageDoc->get_BlackWhiteImage( &bwImage ) );

        // Save bitonal image to file
        CBstr outputFilePath = L"result\\bitonal.png";
        CheckResult( bwImage->WriteToFile( outputFilePath,
        IFF_PngBwPng ) );
  }
```

Since ABBYY uses the same technology base for many different products, the new algorithms developed for the Impact project are already available in Recognition Server 3.0. Some, but not all of them will soon be available in the desktop product, FineReader 12.

For further technical details see the FineReader Engine 10 User's Guide.

1.5. Evaluation

The main purpose of the developed binarisation toolkit was to improve recognition quality. Using visual criteria to measure binarisation quality is in this case not appropriate. Our experience shows that bitonal images that look better to the human eye are not necessarily recognized better than bitonal images obtained from OCR-targeted binarisation. Vice versa, text in images produced by such a specific binarisation process cannot be expected to look perfect to the human eye, but it is usually recognized with a higher degree of accuracy.

The natural way to measure binarisation quality is therefore measuring the recognition quality. Another reason for this evaluation approach is that it is impossible to visualise results of the second binarisation stage which is performed while the engine tries to recognise each part of text many times with different binarisation settings.

Evaluation performed with ground truth data shows that, using the new version 10 of FineReader Engine, 15% less text recognition errors are produced. It has to be noted that this evaluation measures not only improvements in binarisation but also the effects of improvements with regard to the recognition engine (TR3). Nevertheless, binarisation plays a major role in the overall improvement.

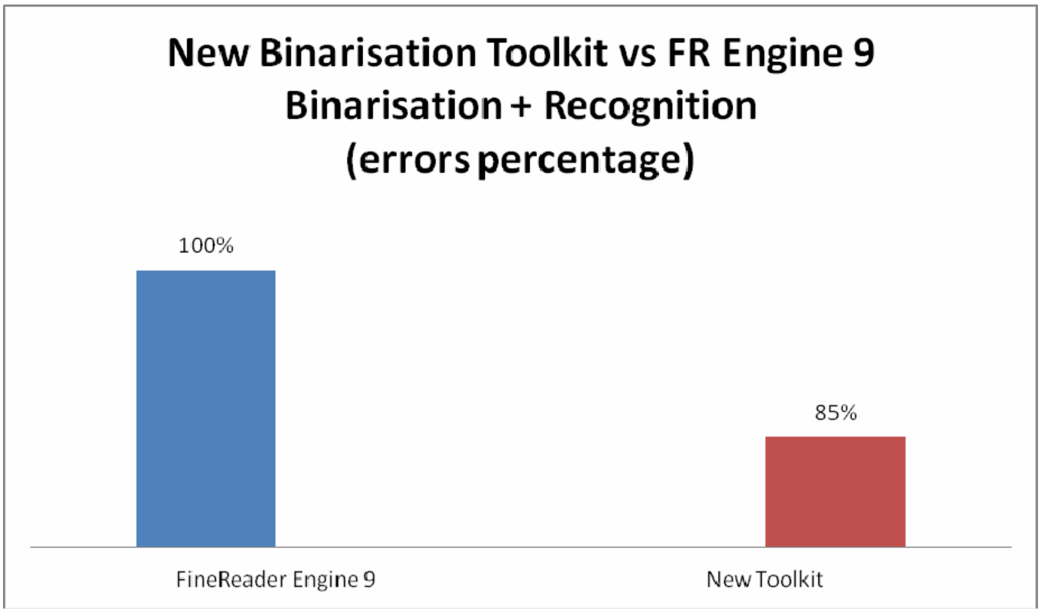


Figure 4: Recognition error rate – FR Engine 9 vs. FR Engine 10

However, it is possible to visualise the results of the first binarisation stage. Figure 5 shows some examples of the new global binarisation in comparison to the old (pre-IMPACT) one:

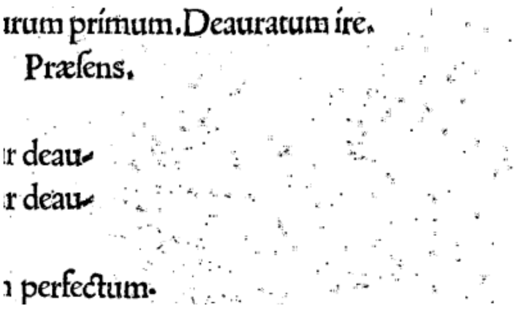
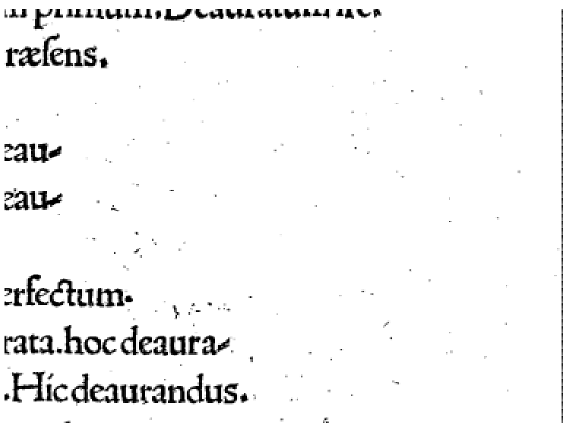
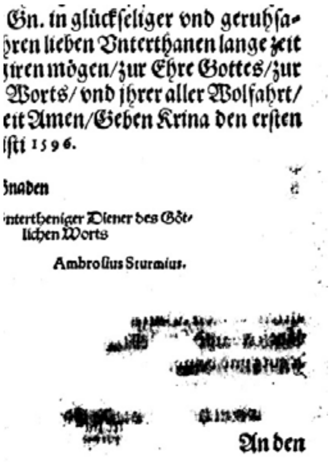
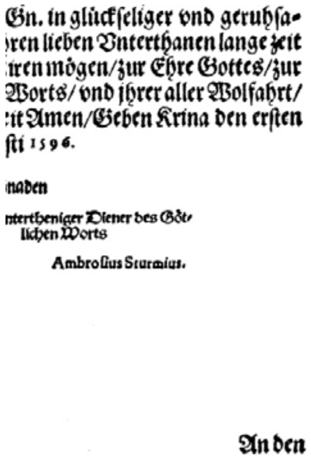
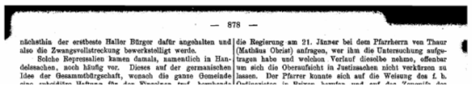
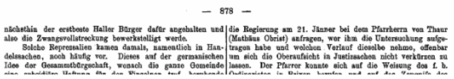
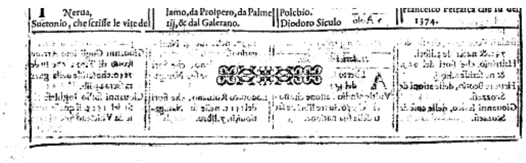
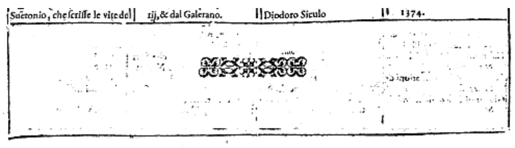
Pre-IMPACT FineReader Engine 9	New toolkit / FineReader Engine 10
	 *Less noise on image*
	 *Less noise on image*
	 *No top shadow*
	 *No bleed-through*

Figure 5: Visual comparison of binarisation results – FR Engine 9 vs. FR Engine 10

Another evaluation was performed in order to measure the quality of the first binarisation stage only. The main difficulty here is that after each modification in the binarisation algorithm, classifiers that perform character recognition are modified to better work with the current binarisation. In order to obtain more reliable results, the following evaluation scheme was implemented:

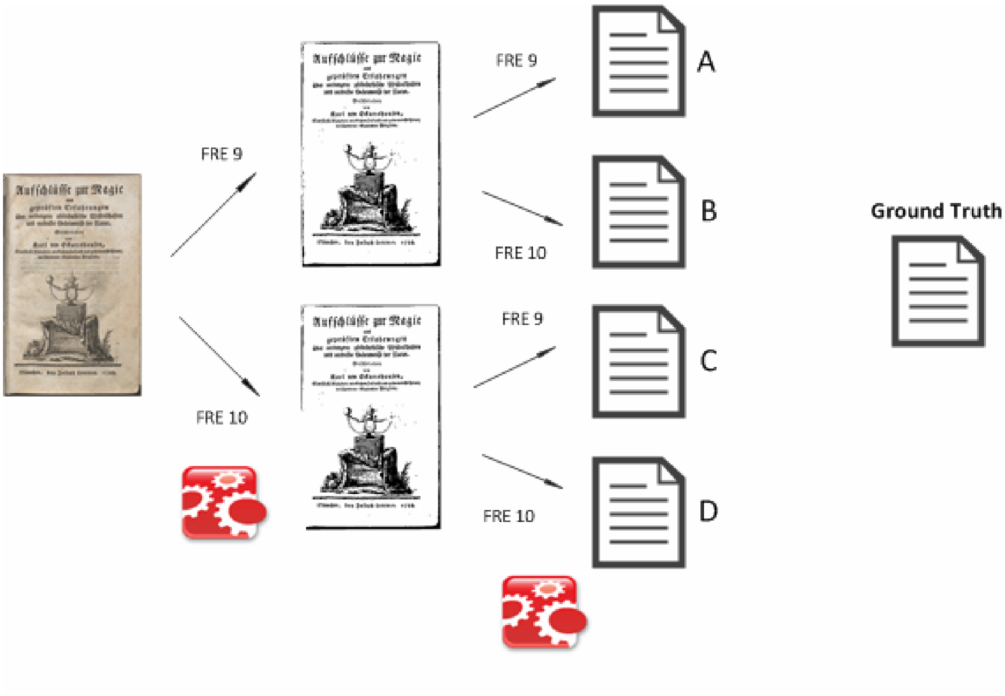


Figure 6: Comparative evaluation approach

Each colour image in a test set was binarised using two methods and thus obtaining two bitonal images. Then, each of these bitonal images was recognized with two recogniser versions. This way, four recognition results were obtained. The results are shown below:

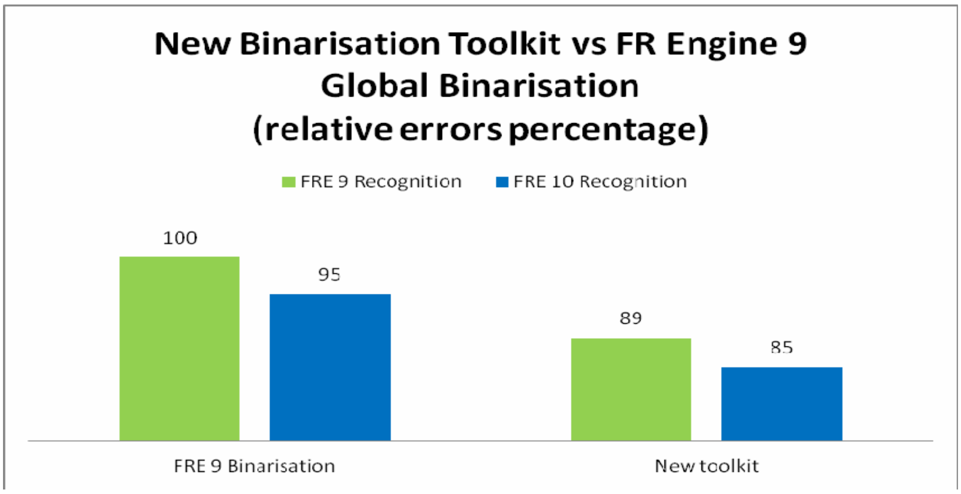


Figure 7: Comparative evaluation results

Concluding, out of 15% improvement in recognition accuracy, about 10% is due to the new global binarisation. Another 5% were achieved by implementing a second binarisation step during the recognition stage and by classifier tuning and improvement.

Address the contribution towards project expectations. Tick the box according to which category your tool falls into:

☒	*Software tool is fit to be put into productive use and is supported by the necessary installation guides*
□	*Software tool can be made available in a productive environment with further development which is clearly defined*
□	*Software tool demonstrates potential functionality and is available in a publicly accessible environment*
□	*Report of findings of research available (for experimental tools only)*

1.6. License and IPR protection

ABBYY is not going to patent any work which is done within IMPACT project.

All parts of the toolkit will be available as part of the FineReader Engine product line. Licensing of the toolkit is the same as actual licensing of FineReader Engine.

2. Noise and Artefacts Removal Toolkit

2.1. Partner

NCSR

2.2. Deliverable

D-TR1(b). Border Removal Toolkit

2.3. Background

Document images are usually scanned pages from books, periodicals or newspapers. Approaches proposed for document segmentation and character recognition usually consider ideal scanned images without noise. However, there are many factors that may generate imperfect document images. When a page of a book is scanned, text from an adjacent page may also be captured into the current page image (see Figure 8). These unwanted regions are called “noisy text regions”. Additionally, whenever a scanned page does not completely cover the scanner setup image size, there will usually be black borders in the image (see Figure 8). These unwanted regions are called “noisy black borders”. All these problems influence the performance of segmentation and recognition processes. Since the page segmentation algorithms report noisy text regions as text-zones, the OCR accuracy decreases in the presence of noisy regions, because the OCR system usually outputs several extra characters in these regions. The goal of border detection is to find the text region, ignoring the noisy text and black borders.

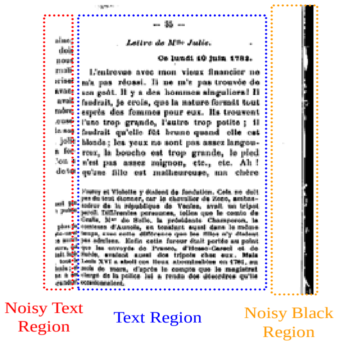


Figure 8: Example of an image with noisy black border, noisy text region and text region.

Methodologies in the literature

The most common approach to eliminate marginal noise is to perform document cleaning by filtering out connected components based on their size and aspect ratio [1],[2]. However, when characters from the adjacent page are also present, they usually cannot be filtered out using these features alone. There are only few techniques in the bibliography for page borders detection.

Le and Thoma [3] propose a method for border removal which is based on classification of blank, textual and non-textual rows and columns, location of border objects, and an analysis of projection profiles and crossing counts of textual squares. Their approach uses several heuristics and it is also based on the assumption that the page borders are very close to edges of images and borders are separated from image contents by a white space. However, this assumption is often violated.

Fan et al. [4] propose a method for removing noisy black regions overlapping the text region, but do not consider noisy text regions. They propose a scheme to remove the black borders of scanned documents by reducing the resolution of the document image. This approach consists of two steps, marginal noise detection and marginal noise deletion. The block diagram is illustrated in Figure 9. Marginal noise detection consists of three steps: (i) resolution reduction, (ii) block splitting, and (iii) block identification. The flowchart of marginal noise detection is shown in Figure 10. Marginal noise detection makes the textual part of the document disappear leaving only blocks to be classified either as images or borders by a threshold filter. Marginal noise has to be deleted after it has been detected. The deletion process should be performed on the original image instead of the reduced image .The block classification is used to segment the original image, removing the noisy black borders.

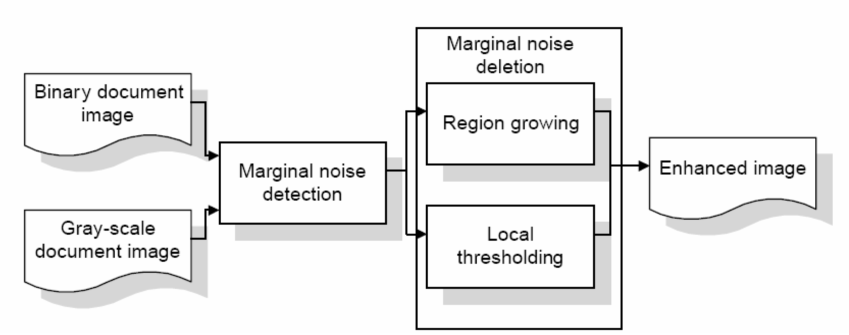


Figure 9: Block diagram of Fan et al. method [4].

Fig. 2. The block diagram of the system.

Fig. 3. The flowchart of marginal noise detection.

literate small blocks, whereas a small reduc-
leave characters as pepper-and-salt noise. The
of appropriate reduction rate is important in
he resolution reduction task.

an example as shown in Fig. 4 to demonstrate the proce
where the reduction rate is chosen to be 6.

2.2. Block splitting

Figure 10: Flowchart of marginal noise detection.

Avila et al. [5] propose the invading and non-invading border algorithms which work as “flood–fill” algorithms. The invading algorithm assumes that the noisy black border does not invade the black areas of the document. It moves outside the noisy surrounding borders towards the document. In the case that the document text region is merged to noisy black borders the whole area, including the part of text region is flooded and removed. Contrarily, the non-invading border algorithm assumes that noisy black borders merge with document information. In order to restrain flooding in the whole connected area it takes into account two parameters related to the nature of the documents. These parameters are the maximum size of a segment belonging to a document and the maximum distance between lines.

Also, Avila et al. [6] propose an algorithm based on “flood-fill” component labelling and region adjacency graphs for removing noisy black borders in monochromatic images. The proposed algorithm encompasses five steps: (i) flooding, (ii) segmentation, (iii) component labelling, (iv) region adjacency graph generation and (v) noise border removal.

Commercial products

Border removal functionality can be found in the following commercial products:

- Book Restorer (i2S) (http://www.i2s-bookscanner.com/) (image restoration software – see Figure 11)
- WiseBook (CSoft) (http://www.csoft.com/products/wisebook/) (book scanning software –see Figure 12)
- ScanFix (accusoft pegasus) (http://www.accusoft.com/scanfix.htm) (image cleanup SDK)

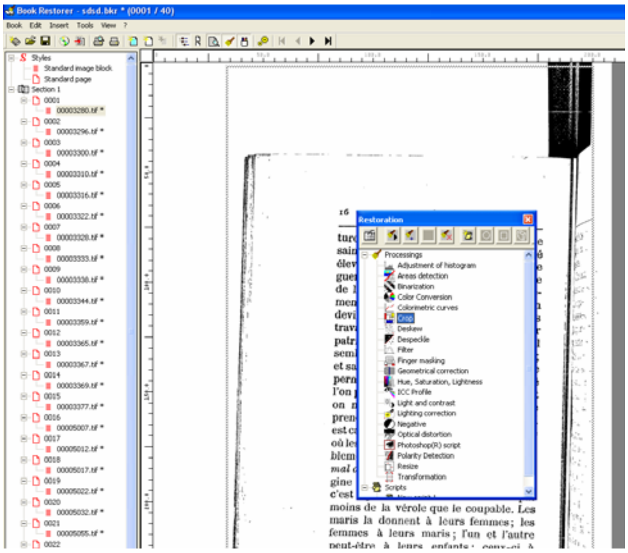
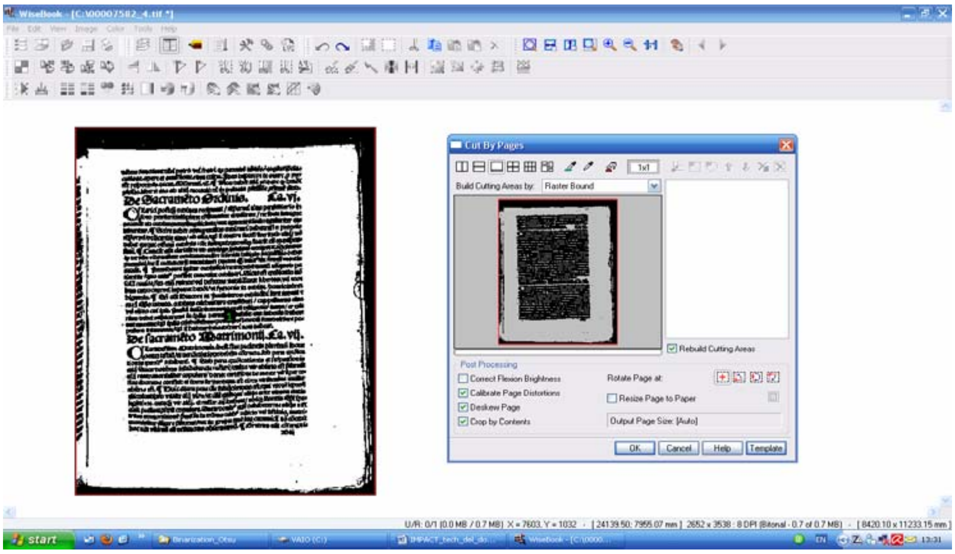


Figure 11: Book Restorer (i2S) - Image restoration software


WiseBook - [C:\00007582_4.tif *]
File Edit View Image Color Tools Help
Cut By Pages
OK
Cancel
Help
Template
start

Build Cutting Areas by: Raster Bound

☑ Rebuild Cutting Areas

Post Processing

□ Correct Flexion Brightness

☑ Calibrate Page Distortions

☑ Deskew Page

☑ Crop by Contents

Rotate Page at:

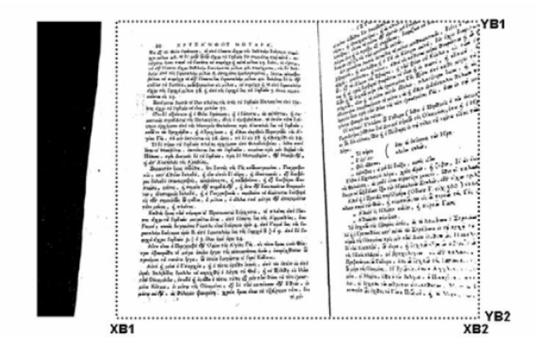
□ Resize Page to Paper

Output Page Size: [Auto]

Figure 12: WiseBook (CSoft) - Book scanning software

2.4. Outline of functionality

Our methodology detects and removes noisy black borders as well as noisy text regions. It is based on projection profiles combined with a connected component labelling process. Signal cross-correlation is also used in order to verify the detected noisy text areas. First, we detect and remove noisy black borders (vertical and horizontal) of the image. Our aim is to calculate the limits of text regions (XB1, XB2, YB1 and YB2) (see Figure 13). The flowchart for noisy black borders detection and removal is shown in Figure 13: The limits of text regions. Figure 14. In order to achieve this, we first proceed to an image smoothing, then calculate the starting and ending offsets of borders and text regions and then calculate the borders limits. The final clean image without the noisy black borders is calculated by using the connected components of the image. The main modules of the proposed technique are as follows:



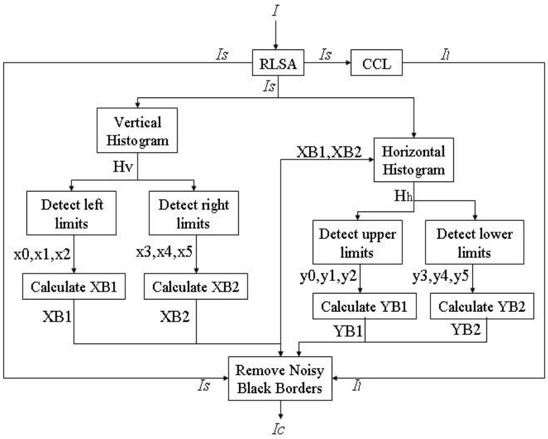


Figure 13: The limits of text regions.

Figure 14: Flowchart for noisy black borders detection and removal.

<u>RLSA</u>: Horizontal and vertical smoothing with the use of the Run Length Smoothing Algorithm (RLSA). RLSA examines the white runs existing in the horizontal and vertical direction. For each direction, white runs with length less than a threshold are eliminated. The empirical value of horizontal and vertical length threshold is 4 pixels.

<u>CCL (Connected Component Labeling)</u>: Calculate the connected components of the image.

<u>Vertical Histogram</u>: Calculate vertical histogram of the image, which is the sum of black pixels in each column.

<u>Detect left limits</u>: Detect vertical noisy black borders (x_0, x_1, x_2) in the left side of the image using the vertical histogram (see Figure 15).

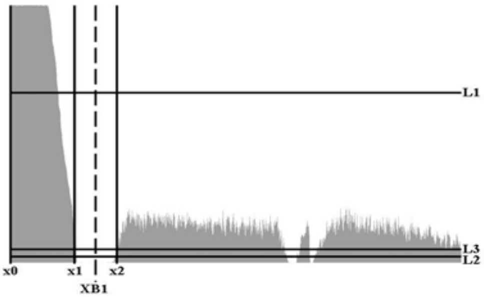


Figure 15: Projections of an image and left limits detection.

Calculate XB1: Calculate left limit (XB1) of text regions (see Figure 13) as follows:

$$XB1 = \begin{cases} 0 & \text{if } x_0 = -1 \\ x_0 + (x_1 - x_0)/2 & \text{if } x_2 = -1 \\ x_1 + (x_2 - x_1)/2 & \text{if } x_2 \neq -1 \end{cases} \qquad (1)$$

A similar process is applied in order to detect the vertical noisy black border of the right side of the image as well as the right limit XB2 of text regions.

Horizontal Histogram: Calculate horizontal histogram which is the sum of black pixels in each row at XB1 and XB2 limits.

A similar process as for the vertical noisy black borders is applied in order to detect the horizontal noisy black borders as well as the upper (YB1) and bottom (YB2) limits of text regions.

Remove Noisy Black Borders: All black pixels that belong in a connected component which includes at least one pixel that is out of limits are transformed in white.

Once the noisy black borders have been removed, we proceed to detect noisy text regions of the image. Our aim is to calculate the limits of the text region (XT1 and XT2) (see Figure 16). The flowchart for noisy text regions detection and removal is shown in Figure 16: The limits of the text region. Figure 17. We first proceed to an image smoothing in order to connect all pixels that belong to the same line. Then, we calculate the vertical histogram in order to detect text zones. Finally, we detect noisy text regions with the help of the signal cross-correlation function. The main modules of the proposed technique are described in detail as follows.

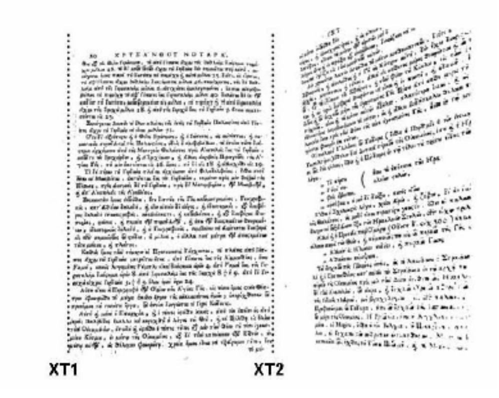


Figure 16: The limits of the text region.

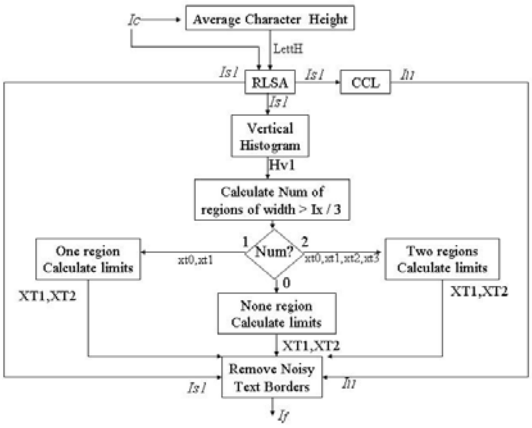

Ic
Average Character Height
LettH
Is1
RLSA
Is1
CCL
Ih1
Is1
Vertical Histogram
Hv1
Calculate Num of regions of width > Ix / 3
1
Num?
2
xt0,xt1
xt0,xt1,xt2,xt3
One region Calculate limits
Two regions Calculate limits
XT1,XT2
XT1,XT2
0
None region Calculate limits
XT1,XT2
Is1
Remove Noisy Text Borders
Ih1
If

Figure 17: Flowchart for noisy text regions detection and removal.

Average Character Height: The average character height (LettH) for the document image is calculated.

RLSA: Horizontal and vertical smoothing with the use of the Run Length Smoothing Algorithm (RLSA) by using dynamic parameters which depend on average character height. Our aim is to connect all pixels that belong to the same line.

CCL (Connected Component Labeling): Calculate the connected components of the image.

Vertical Histogram: Calculate vertical histogram of the image.

Calculate Number of regions: We detect the regions which are split by a blank area using the vertical histogram and also have a width greater than $I_x / 3$ (I_x denotes the image width). Let us suppose that two regions have been found and let xt_0 , xt_1 and xt_2 , xt_3 denote the regions' limits, as shown in Figure 18.

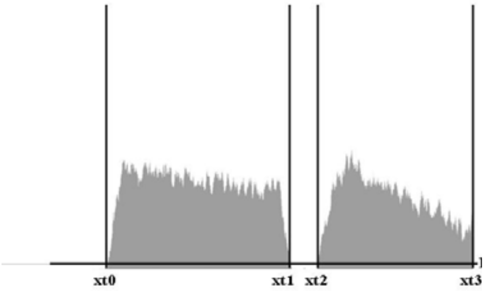


Figure 18: Projections of the image and text regions detection.

Two regions-Calculate Limits: We examine if one of these regions is a noisy text region using the signal cross-correlation. If the signal cross-correlation is greater than 0.5 we remove the region as a noisy text region. If both regions have a signal cross-correlation greater than 0.5 we remove the region which has the greater value.

One region-Calculate Limits: We examine if the noisy text region and the text region are very close to each other without leaving a blank line between them. If the width of the region is less than 70% of the image width we consider that we do not have a noisy text region. Otherwise, we divide it into eight regions and calculate the signal cross-correlation for each region.

No region-Calculate Limits: In this case, the text region consists of two or more columns and we try to locate and separate them from the noise text regions using the signal cross-correlation.

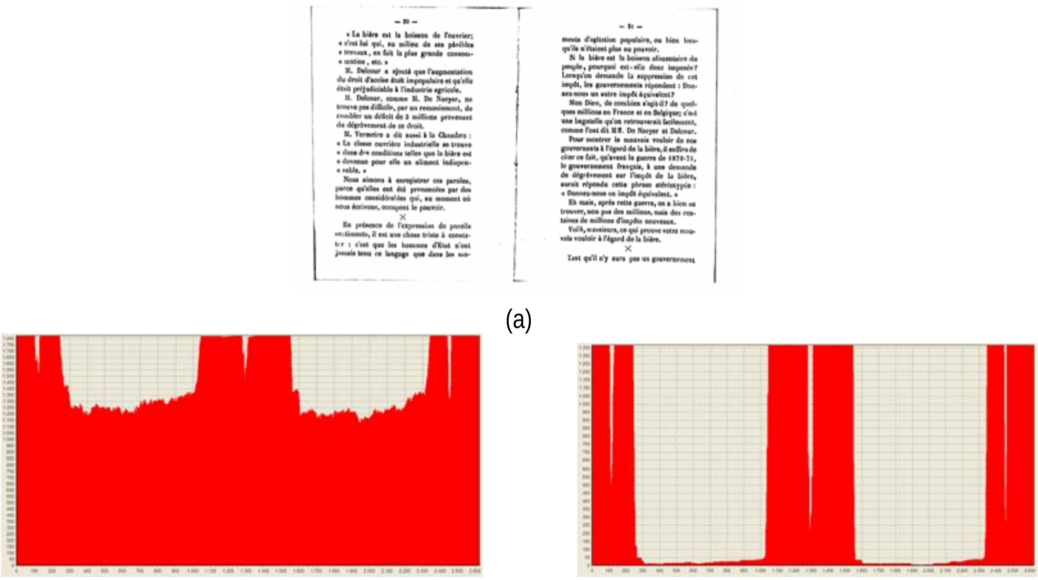
Remove Noisy Text Region: All black pixels that belong in a connected component which does not include at least one pixel in the limits are transformed in white.

The new version of the border removal toolkit, except border removal process, also detects the optimal page frames of double page document images and splits the image into the two pages as well as removes noisy borders. Page split methodology consists of three distinct steps. At a first step, a pre-processing which includes noise removal and image smoothing is applied. At a next step, the vertical zones of the two pages are detected. Finally, the frame of both pages is detected after calculating the horizontal zones for each page.

In order to detect the vertical zones we focus on the white pixels of the image and introduce the vertical white run projections *HV()* which have been proved efficient for detecting vertical zones of text areas. The motivation for proposing these projections is the need to stress the existence of long vertical white runs in the image. The vertical white run projections *HV()* are defined as follows:

$$HV(x) = \frac{\sum_{j=1}^{wv_i} (y_{j2} - y_{j1})^2}{(1-2*a)*Iy} \qquad (2)$$

where wv_i is the number of white runs (i,y_{j1})-(i,y_{j2}) for row $x=i$ in the range of $y=a*I_y$… (1-a)*I_y , I_y denotes the image height and $HV(x) \in$ [0… (1-2*a)*I_y]. An example of the vertical white run projections compared to classical vertical white pixel projections is given in Figure 19. In this figure it is demonstrated that using white run projections we can better discriminate text from non-text vertical zones. We can safely consider that if $HV(x)<(1-2*a)I_y/2$ then the corresponding image column may contain page information. Also, if we have consecutive n image columns that satisfy this equation with $n>b*I_x$ then a vertical page zone is detected. Although detecting two vertical page zones is the most common case, we examine all the cases where more than two, just one or no vertical page zones are detected.



(a)

(b) (c)

Figure 19: (a) Binary image (b) vertical white pixel projections and (c) vertical white run projections.

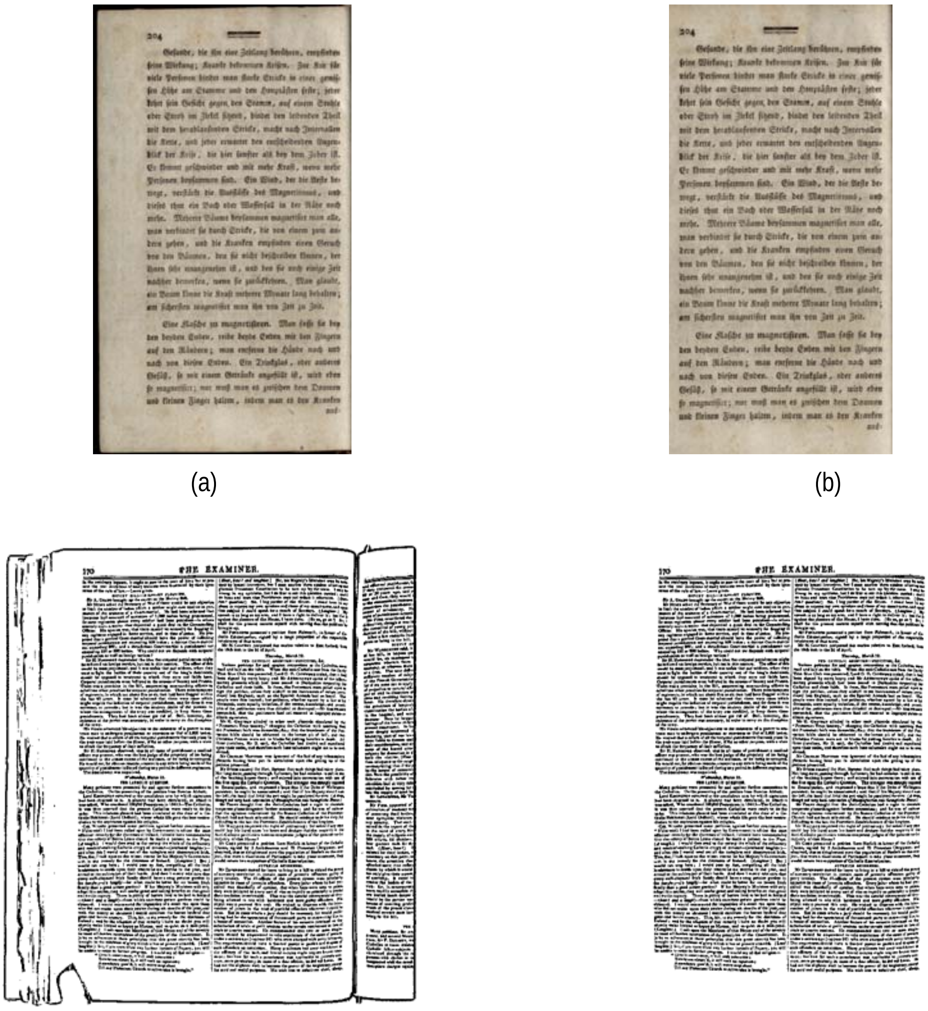
Once the vertical page zones have been detected, a similar process is applied in order to detect the horizontal page zones.

Examples of the developed border removal and page split methodology are given in Figure 20 and Figure 21, respectively.

The final version of the border removal toolkit (v.4) was delivered in the form of a console application for Windows:

```
Border_Detection_v4 [0/1] [in] [out1] [out2]
```

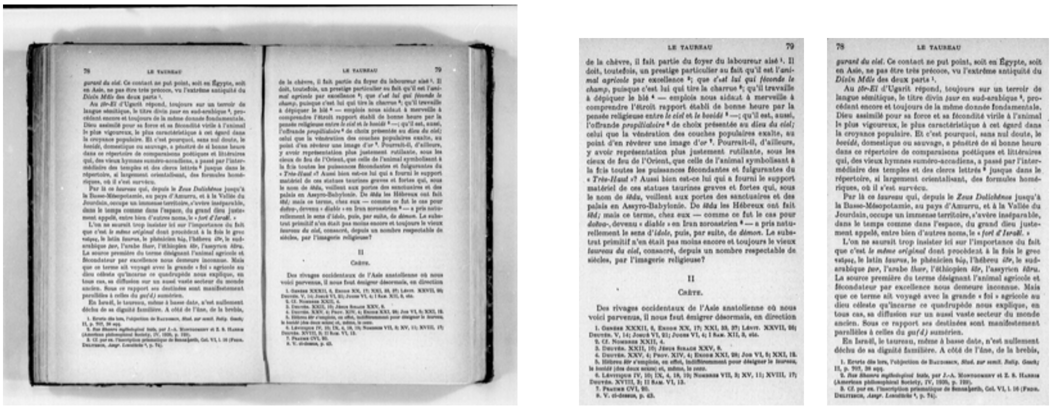
If the first parameter is 0 then border removal is applied otherwise border removal and page split are applied. Performance enhancements have been added to the current version and also it has been tested on over 40000 images. Moreover, this version is able to process binary, grayscale or colour images.



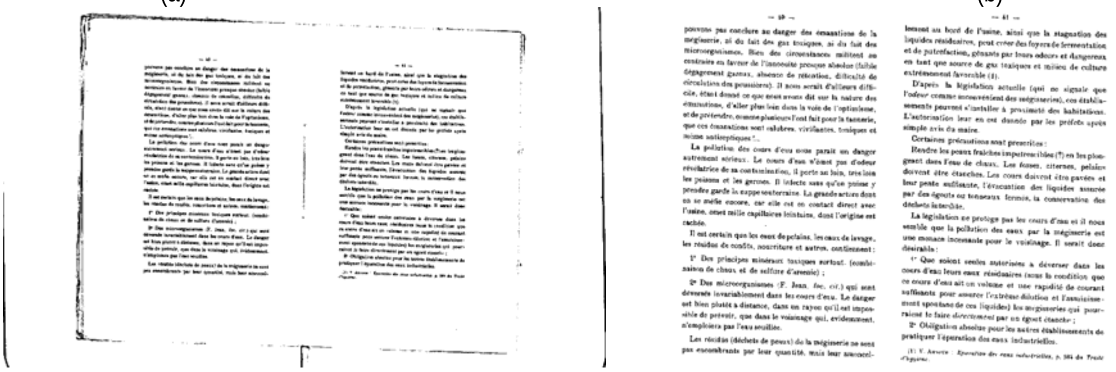
(a) (b)

(c) (d)

Figure 20: Border removal examples; (a),(c) original images; (b),(d) images after border removal.



(a) (b)



(c) (d)

Figure 21: Page split examples; (a),(c) original images; (b),(d) images after page frame detection.

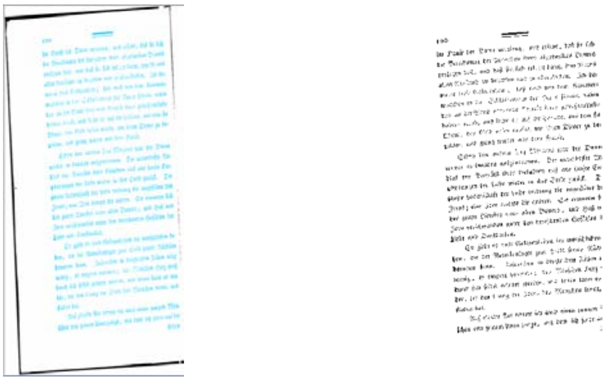
2.5. Evaluation

For the evaluation of the developed border removal algorithm we manually mark the correct text region in the original b/w images in order to create the ground truth set (see Figure 22(a)). The performance evaluation is based on a pixel based approach and counts the pixels at the correct text region and the result image after border removal (see Figure 22(b)). Let *G* be the set of all pixels inside the correct text region in ground truth, *R* the set of all pixels inside the result image and *T(s)* a function that counts the elements of set *s*. We calculate the precision and recall as follows:

$$Precision = \frac{T(G \cap R)}{T(R)} \qquad (3)$$

$$Recall = \frac{T(G \cap R)}{T(G)} \qquad (4)$$

In our example (see Figure 22) Precision is 100% because all the black borders have been removed and Recall is 94% because text which belongs to the correct text region has been cropped.



(a) (b)

Figure 22: (a) Marked text region (ground truth); (b) result after borders removal

Concerning the border removal algorithm, we used a set of 38718 (SET-A) randomly selected historical images, which contains both images with noisy black border and images without noisy borders. For the sake of clarity, we also present the evaluation results using only the images with noisy black border (SET-B). For comparison purposes, we applied at the same set the state-of-the art method [3] (D.X. Le) as well as the commercial products ScanFix, BookRestorer and WiseBook. Table 1 and

Table 2 illustrate the average precision, recall and FM of all methods. As it can be observed,

- The proposed border removal method outperforms all the others methods and achieves FM= 98.93% for SET-A and FM= 98.54% for SET-B.
- Best state-of-the-art methodology found to be the D.X Le method which achieves FM= 97.30% for SET-A and FM= 95.62% for SET-B.

Table 1: Border Removal - Evaluation results using SET-A

		BL	BNE	BNF	BSB	JSI	NLB	ONB	TOTAL
	#images	3632	11126	12251	4784	4430	706	1789	38718
IMPACT	Prec (%)	99.49	99.89	98.88	98.10	98.91	99.86	97.29	99.08
	Rec (%)	98.83	99.26	99.40	96.07	99.06	99.73	97.82	98.79
	FM (%)	99.16	99.58	99.14	97.07	98.99	99.79	97.55	98.93
D.X Le	Prec (%)	94.98	99.68	98.67	97.70	97.35	99.80	97.19	98.30
	Rec (%)	99.31	90.65	99.24	95.58	99.21	99.81	99.19	96.63
	FM (%)	97.10	94.95	98.26	96.63	98.27	99.80	98.18	97.30
BookRestorer	Prec (%)	91.13	96.88	98.08	97.29	94.50	99.79	95.12	96.47
	Rec (%)	99.56	91.57	99.77	97.43	99.40	99.85	99.61	97.06
	FM (%)	95.16	94.15	98.91	97.36	96.89	99.82	97.31	96.76
WiseBook	Prec (%)	86.93	88.57	91.20	95.76	90.69	99.46	80.37	90.20
	Rec (%)	98.37	99.47	99.10	96.40	97.29	98.45	98.63	98.56
	FM (%)	92.30	93.71	94.99	96.08	93.87	98.95	88.57	94.20
ScanFix	Prec (%)	81.65	92.87	91.29	95.62	91.00	99.24	84.52	91.17
	Rec (%)	94.97	98.66	98.66	97.81	95.66	80.81	96.98	97.46
	FM (%)	87.81	95.68	94.83	96.70	93.27	89.08	90.32	94.21

Table 2: Border Removal - Evaluation results using SET-B

		BL	BNE	BNF	BSB	JSI	NLB	ONB	TOTAL
	#images	1631	7543	7677	2417	1416	315	1384	22383
IMPACT	Prec (%)	98.94	99.88	98.29	96.86	98.01	99.98	96.63	98.62
	Rec (%)	98.18	99.27	99.26	93.14	99.15	99.87	98.24	98.46
	FM (%)	98.56	99.57	98.77	94.96	98.58	99.92	97.43	98.54
D.X Le	Prec (%)	88.89	99.58	97.98	96.08	93.20	99.85	96.48	97.28
	Rec (%)	99.05	86.64	98.86	91.53	99.09	99.97	99.06	94.01
	FM (%)	93.70	92.66	98.42	93.75	96.05	99.91	97.75	95.62
BookRestorer	Prec (%)	80.30	95.46	97.00	95.27	84.26	99.83	93.76	94.11
	Rec (%)	99.36	93.02	99.68	95.22	99.56	99.96	99.62	96.92
	FM (%)	88.82	94.22	98.32	95.24	91.27	99.89	96.60	95.50
WiseBook	Prec (%)	70.98	83.24	86.10	92.24	72.44	99.18	74.77	83.32
	Rec (%)	99.38	99.49	99.58	95.19	98.36	98.61	99.09	98.94
	FM (%)	82.81	90.64	92.35	93.69	83.43	98.89	85.23	90.46
ScanFix	Prec (%)	59.23	89.57	86.23	91.96	73.38	99.04	80.14	85.00
	Rec (%)	95.42	98.78	99.03	96.54	98.55	80.61	97.67	98.04
	FM (%)	73.09	93.95	92.19	94.19	84.12	88.88	88.04	91.05

Concerning page split algorithm, we used a set of 3467 double page historical images. Table 3 illustrates the average precision, recall and FM of the developed page split methodology. As it can be observed, it achieves FM= 95.09%.

Table 3: Page Split - Evaluation results

		BL	BNF	BSB	JSI	TOTAL
	#images	2171	458	305	533	3467
IMPACT	Prec (%)	92.11	98.97	94.36	84.47	92.04
	Rec (%)	98.15	98.99	99.57	97.90	98.35
	FM (%)	95.03	98.98	96.89	90.69	95.09

Address the contribution towards project expectations. Tick the box according to which category your tool falls into:

☒	*Software tool is fit to be put into productive use and is supported by the necessary installation guides*
□	*Software tool can be made available in a productive environment with further development which is clearly defined*
□	*Software tool demonstrates potential functionality and is available in a publicly accessible environment*
□	*Report of findings of research available (for experimental tools only)*

2.6. License and IPR protection

This part of work is owned by the National Centre for Scientific Research "Demokritos", Greece (NCSR). After the project ends, individual licensing agreements for commercial and non-commercial use will be required.

2.7. References

[1] Breuel, T.M.: Two geometric algorithms for layout analysis. In: Document Analysis Systems, Princeton, NY (Aug. 2002) 188–199

[2] O Gorman, L.: The document spectrum for page layout analysis. IEEE Trans. On Pattern Analysis and Machine Intelligence 15(11) (Nov. 1993) 1162–1173

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

[4] *Kuo-Chin Fan, Yuan-Kai Wang, Tsann-Ran Lay,* Marginal Noise Removal of Document Images, Pattern Recognition, 35(11), pp. 2593-2611, 2002.

[5] *B.T. Avila and R.D. Lins,* A New Algorithm for Removing Noisy Borders from Monochromatic Documents, Proc. of ACM-SAC'2004, pp 1219-1225, Cyprus, ACM Press, March 2004.

[6] B*.T. Avila, R.D. Lins,* Efficient Removal of Noisy Borders from Monochromatic Documents, ICIAR 2004, LNCS 3212, pp. 249-256, 2004.

3. Geometrical Defect Correction Toolkit

3.1. Partner

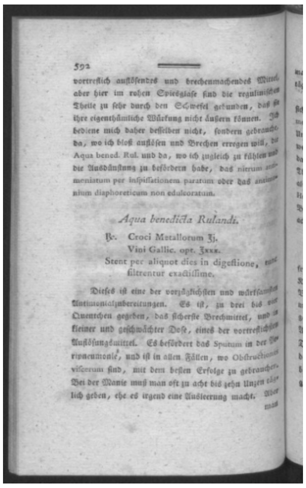
NCSR

3.2. Deliverable

D-TR1(c). Document Image Dewarping Toolkit

3.3. Background

Document image acquisition by a flatbed scanner or a digital camera often results in several unavoidable image distortions (see Figure 23) due to the form of printed material (e.g. bound 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.



592

vortreflich auflöſendes und brechenmachendes Mittel, aber hier im rohen Spiesglaſe ſind die reguliniſchen Theile zu ſehr durch den Schwefel gebunden, daß ſie ihre eigenthümliche Würkung nicht äußern können. Ich bediene mich daher deſſelben nicht, ſondern gebrauche da, wo ich bloß auflöſen und Brechen erregen will, die Aqua bened. Rul. und da, wo ich zugleich zu kühlen und die Ausdünſtung zu befördern habe, das nitrum antimoniatum per inſpiſſationem paratum oder das antimonium diaphoreticum non edulcoratum.

Aqua benedicta Rulandi.

℞. Croci Metallorum ℥j.
Vini Gallic. opt. ℥xxx.
Stent per aliquot dies in digeſtione, tunc filtrentur exactiſſime.

Dieſes iſt eine der vorzüglichſten und würkſamſten Antimonialzubereitungen. Es iſt, zu drei bis vier Quentchen gegeben, das ſicherſte Brechmittel, und in kleiner und geſchwächter Doſe, eines der vortreflichſten Auflöſungsmittel. Es befördert das Sputum in der Peripneumonie, und iſt in allen Fällen, wo Obſtructiones viſcerum ſind, mit dem beſten Erfolge zu gebrauchen. Bei der Manie muß man oft zu acht bis zehn Unzen täglich geben, ehe es irgend eine Ausleerung macht. Aber man

Figure 23: Example of document image captured by a flatbed scanner.

Existing methodologies in the literature

Over the last decade, many different techniques have been proposed for document image rectification [1] and they can be classified into two main categories based on (i) 3D document shape reconstruction [2-9] and (ii) 2D document image processing [10-22]. Techniques of the former category obtain the 3D information of the document image using special setup or reconstruct the 3D 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 2D information.

Rectification techniques based on 3D document shape reconstruction rely upon extraction of the 3D information of the document and they can be further divided into two subcategories. Techniques of the first subcategory obtain the 3D 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 3D 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 and use the skeleton of horizontal text lines to help estimate the model parameters. 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 3D 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 in which the x-to-y sampling ratio is one and the principal point is located at the image centre. Finally, Tan et al. [8] and L. Zhang et al. [9] are based on a shape-from-shading formulation in order to reconstruct the 3D shape of the document's surface. These techniques require knowledge of lighting which in most of the cases is unknown.

Rectification techniques based on 2D document image processing rely on the use of 2D information available in document images. The majority of these rectification techniques 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. [10] use an active contour network based on an analytical model with cubic B-splines that have been proved more accurate than Bezier curves. They propose an automation of the initialization by using an approach based on a particle system which is restricted to be very close to the desired solution. Furthermore, this approach is not efficient in the case of inhomogeneous line spacing. Wu and Agam [11] have built a mesh of the warped image using a non-linear curve for each text line. The curves are fitted to text lines by tracking the character boxes in the text lines. This method 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 either. In [12], L. Zhang and Tan rely on a Gordon surface model constructed from the text lines. The text lines are represented using natural cubic splines interpolating a set of points extracted from connected component analysis. They assume that the book spine is found along iso-parametric lines. Ezaki et al. [13] use cubic splines not only to model the distorted text lines but also the space between them. For more accurate results, a vertical division of a document image into some partial document images is applied. This method uses complex computations and also the line-warping model is not so accurate. Finally, Mischke and Luther [14] detect the distorted text lines using the centroid of each connected component's bounding box. Each distorted text line is polynomially approximated and a dense source mesh is constructed. The rectified image is created by applying a classical bilinear transformation. They 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.

There are also rectification techniques that rely on text line detection and emphasize on baseline finding. Ulges et al. [15] rely on a priori layout information and apply a line-by-line dewarping of the observed paper surface. They 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 and spacing between words is not large. Lu et al. [16] restore the document by dividing images into multiple quadrilateral patches where text is considered to lie on a planar surface. Document partitioning is implemented through 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-based languages. In [17], Schneider et al. use local orientation features extracted by text line baselines to interpolate a vector field from which a warping mesh is derived. The image is corrected by approximating the non-linear distortions with multiple linear projections. 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. [18] after detecting text lines determine their upper and lower baselines using ridges based coupled-snakes model. Then, distortions are removed by mapping characters over each curled baseline pair (upper and lower) to its corresponding straight baseline pair. This method is sensitive to large and variety distortions, especially among the same text line. Finally, Fu et al. [19] and Y. Zhang et al. [20] are based on text line detection in order to extract distortion parameters of the document images. Fu et al. assume that the image surface is a cylinder and generate a transformation to flatten the document image. Image surface extraction is based on text line detection. 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. Y. Zhang et al. take a rough text line and character segmentation to estimate the warping direction. Then, a mapping between the original image and the restored image is determined with several pairs of key points while 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.

Finally, some rectification techniques 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 2D content such as document boundaries or known reference points. Brown and Tsoi [21] propose an approach that uses 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 [22] propose a method for document dewarping using a continuous skeletal image representation. Long continuous branches, which define spaces between text lines of the document, are approximated by cubic Bezier curves in order to find a specific deformation of each space and then a whole approximation of the document is built. This method is sensitive to the approximation of vertical borders deformation in text blocks, which diminish accuracy.

Commercial products

A reliable commercial image restoration tool that provides a document image dewarping functionality is the BookRestorer software [23].

3.4. Outline of functionality

The developed rectification methodology uses only 2D information from document images without any dependence on auxiliary hardware or prior knowledge. The flowchart of the developed rectification methodology is shown in Figure 24.The majority of the state of the art techniques [10-20] are based on text line detection at the original distorted document images which is a well-known hard-task. For this purpose, the developed methodology adopts a two step coarse-to-fine rectification strategy. In the coarse rectification step only some specific points are required to model the curved surface projection and therefore potential erroneous detection results do not seriously affect the projection. 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 [12], [13], [14], [19], and [21], which are based on specific patterns or camera setup parameters making them hard to generalize, or use complex computations, the proposed methodology 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 developed 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 document language and it can deal with documents which contain inhomogeneous text line spacing as well as non-text content like pictures, graphs, etc.

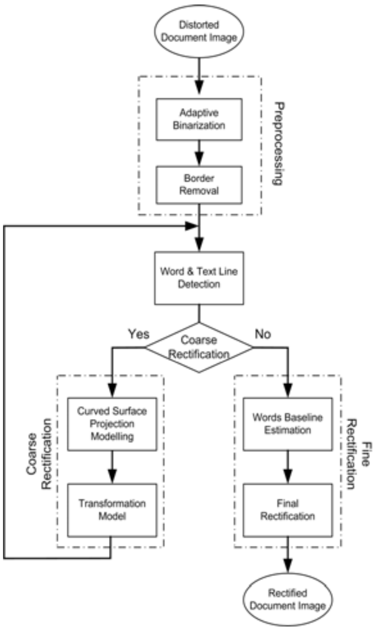


Figure 24: Flowchart of the developed rectification methodology.

Coarse Rectification

In this step, we apply a computationally low cost transformation which addresses the projection of a curved surface to a 2D rectangular area in order to achieve a coarse rectification of the document image. Compared with state of the art techniques [12], [19], and [21], which also use boundaries to delimit the required dewarping, our approach does not use any physical pattern or global model (e.g. cylinder) for the distortion of the document image. Methods [12] and [21] use particular types of interpolation based on Gordon surface model [24] and bi-linearly blended Coons [25], respectively. In our method, we create a correspondence between the points of the two curved line segments and the top and bottom area, upon which the mapping from the projection of a surface to a rectangle is applied. Our primary aim is to restore the large distortions of the document image, so that a rough rectification should be achieved. The rectified outcome will be given as an input in a next step (fine rectification) in order to restore all local distortions of the document image and achieve an optimal rectification of the document image.

Word and text line detection has been applied at the original distorted documents before the coarse rectification step, 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 won't use each detected text line to correct the distortions of the document. 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 $A(x_1, y_1)$, $B(x_2, y_2)$, $C(x_3, y_3)$ and $D(x_4, y_4)$ denote the dominant corner points of the projection of the curved surface (see Fig. 2).

First, the straight lines *AD* and *BC* which correspond to the left and right text boundaries are estimated. For this purpose, the start and end points of each text line are detected and the short text lines are excluded using the average length of text lines. Our aim is to retain the most representative text boundaries of the document and eliminate the short text lines such as titles, marginal text, math types, etc., thus we have a better estimation of the straight lines *AD* and BC (see Figure 25).

Next, we estimate the curved lines *AB* and *DC* which correspond to the top and bottom text lines. In order to select appropriate text lines of the document with representative deformation we select the top and bottom text lines which take part in the calculation of the straight lines *AD* and *BC* (see Figure 25). Using the upper and bottom points, respectively, of the selected text lines the coefficients of 3rd degree polynomial are calculated.

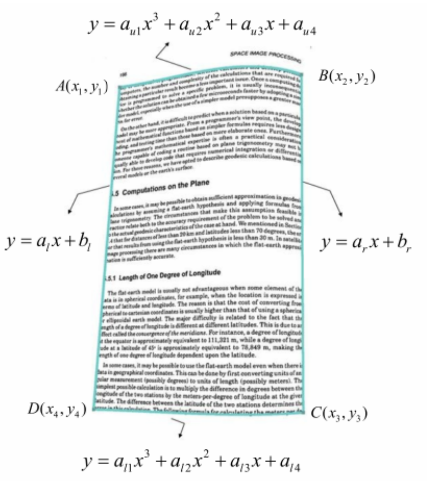

$y = a_{u1}x^3 + a_{u2}x^2 + a_{u3}x + a_{u4}$
$A(x_1, y_1)$
$B(x_2, y_2)$
$y = a_l x + b_l$
$y = a_r x + b_r$
$D(x_4, y_4)$
$C(x_3, y_3)$
$y = a_{l1}x^3 + a_{l2}x^2 + a_{l3}x + a_{l4}$

Figure 25: Example of modelling the curved surface projection on the plane.

After modelling the projection of the curved surface on the plane delimited by the curved line segments *AB* and *DC* and the straight line segments *AD* and *BC*, our goal is to generate a transformation that maps it to a 2D rectangular area. Let $A'(x_1', y_1')$, $B'(x_2', y_2')$, $C'(x_3', y_3')$ and $D'(x_4', y_4')$ denote the corner points of the rectangular area (see Figure 26). Also, let $|\overset{\frown}{AB}|$ and $|AB|$ represent the arc length and the Euclidean distance between points *A* and *B*. The distinct steps we follow are as follows:

<u>Step 1:</u> Allocate the rectangular area *A'B'C'D'*. The width *W* and the height *H* of it are calculated as follows:

$$W = \min(|\overset{\frown}{AB}|, |\overset{\frown}{DC}|) \text{ and } H = \min(|AD|, |BC|) \quad (1)$$

Step 2: Create a correspondence between the points of curved line segments *AB* and *DC*, expressed by a function $\mathcal{F}$ defined as follows: $\mathcal{F}\,(E(x_u, y_u)) = G(x_l, y_l)$ if $\frac{|\overset{\frown}{AE}|}{|\overset{\frown}{AB}|} = \frac{|\overset{\frown}{DG}|}{|\overset{\frown}{DC}|}$, where $E(x_u, y_u)$ represent a point on curved line segment *AB* and $G(x_l, y_l)$ represent a point on curved line segment *DC*.

Step 3: Let $O(x, y)$ represent a point in the projection of the curved surface which belongs to the line *EG*. We calculate the new position $O'(x', y')$ of $O(x, y)$ in the rectangular area as follows (see Figure 26):

$$x' = x_1' + |A'Z| \text{ and } y' = y_1' + |A'H| \tag{2}$$

where *H* is point $H(x_1', y')$, and *Z* is point $Z(x', y_1')$ and $|A'Z|$, $|A'H|$ are calculated as follows:

$$\frac{|\overset{\frown}{AB}|}{|\overset{\frown}{AE}|} = \frac{W}{|A'Z|} \Rightarrow |A'Z| = \frac{W}{|\overset{\frown}{AB}|}|\overset{\frown}{AE}| \text{ and } \frac{|EG|}{|EO|} = \frac{H}{|A'H|} \Rightarrow |A'H| = \frac{H}{|EG|}|EO| \tag{3}$$

Consequently, we repeat *Step 3* for all points which are inside the projection area of the curved surface.

Step 4: Finally, all the points which are out of the projection area if the curved surface inherit the transformation of the nearest point.

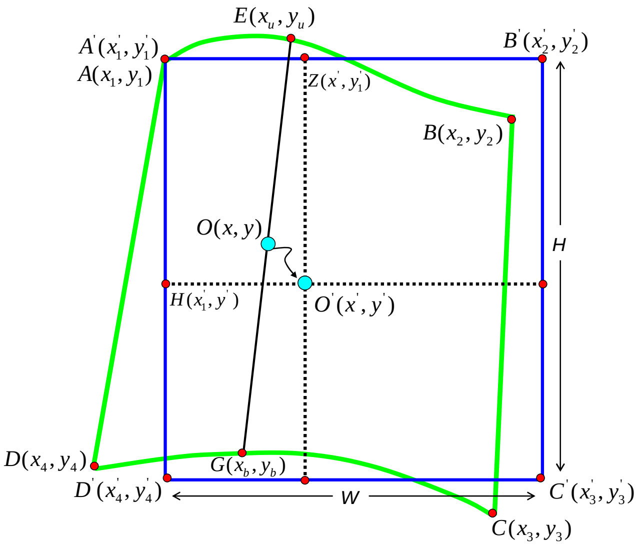
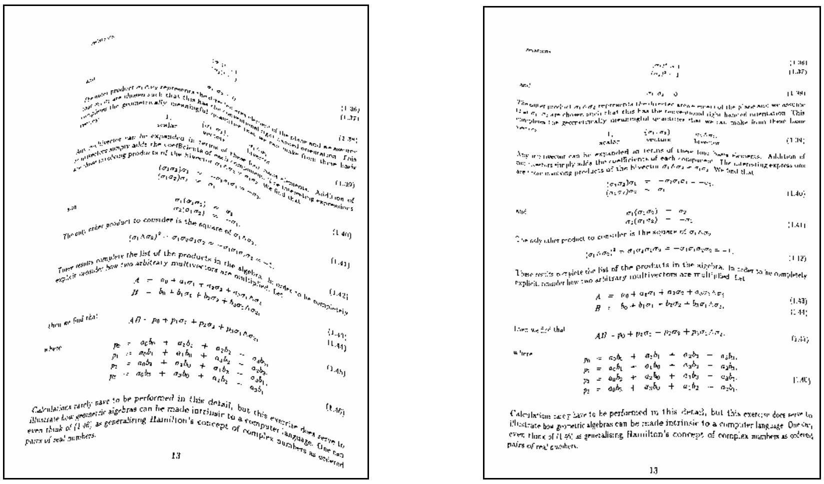


Figure 26: Coarse Rectification: Transformation model: $W = |\overset{\frown}{AB}|$, $H = |AD|$ and $A'(x_1', y_1') \equiv A(x_1, y_1)$.

An example of the coarse rectification step is depicted in Figure 27. As we can observe, the coarse rectification step restores the large distortions of document images and achieves a rough restoration of them. Furthermore, the vertical alignment of the documents is corrected.



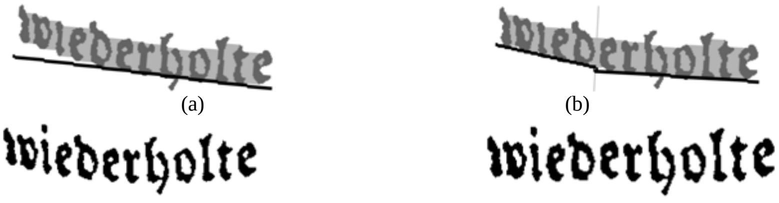
(a) (b)

Figure 27: Example of coarse rectification step: (a) original document image; (b) corresponding rectified document image.

Fine rectification

The second step aims to restore all local distortions of the image, which were not corrected by the first step. At first, we remove the non-text components and then, text line and word detection are applied at the resulting dewarped image after the coarse dewarping stage, thus having very high probability for successful results. All words are detected using a proper image smoothing. Following that, horizontally neighboring words are consecutively linked in order to detect text lines.

Once the words have been detected, we detect the lower and upper baselines which delimit the main body of the words. Several times, words suffer from distortions, so the appropriate baseline cannot be estimated (see Figure 28a). Consequently, only the rotation and translation of the words at the next step would not be enough in order to restore them (see Figure 28c). For this reason, we iteratively split the word (see Figure 28b) and process each part of it independently (see Figure 28d).

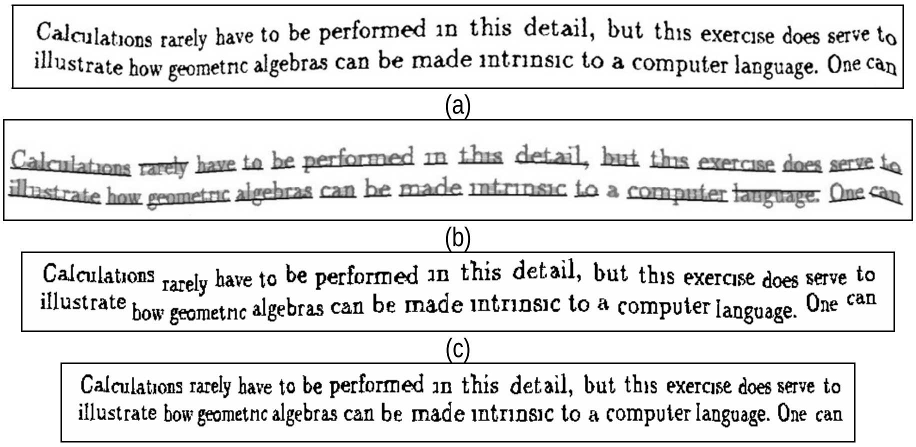


(a) (b)

(c) (d)

Figure 28: Example of baseline estimation and rotation of a distorted word; (a) initial baseline estimation; (b) baseline estimation after word has been splitted; (c), (d) rotation of the word using the baseline of (a) and (b), respectively.

Next, all words are rotated and translated in order to obtain the final rectified image (see Figure 29). First, every word is rotated according to the word's baseline slope. Then, all the words of every text line, except the leftmost are vertically translated in order to restore horizontal alignment. Finally, we add all the components which have been removed as non-text components. In order to achieve this, every pixel inherits the transformation factors of the nearest pixel. Then, we apply a transformation in each component that uses as factors the mean factors of its constituent pixels. An example of coarse and fine document image rectification using the developed methodology is given in Figure 30 while more examples are given in Figure 31. The proposed methodology encounters difficulties when processing a more than one column document or a newspaper as well as when word segmentation fails due to dense layout.



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

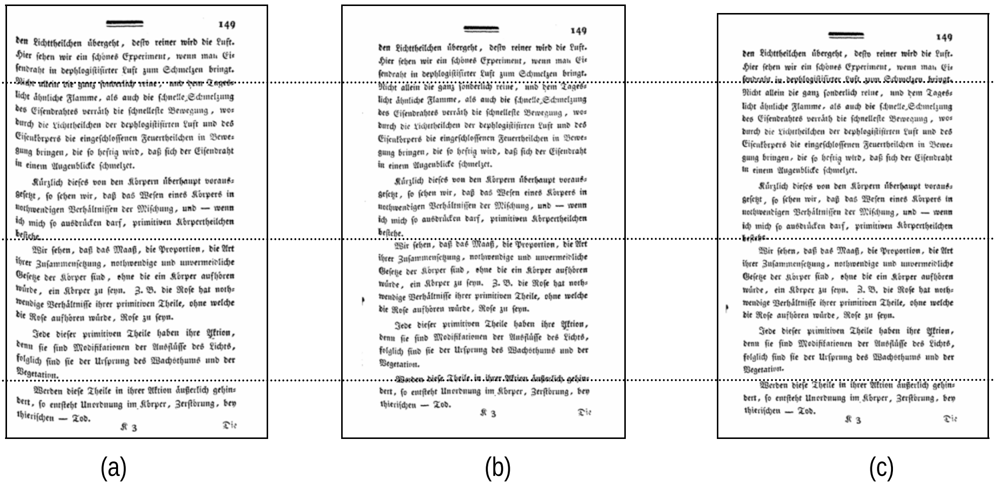
(d)

Figure 29: Example of final rectification; (a) coarse rectified document image; (b) words baseline estimation; (c) rotation of the words; (d) translation of the words.

The final version of the border removal toolkit (v.4) was delivered in the form of a console application for Windows:

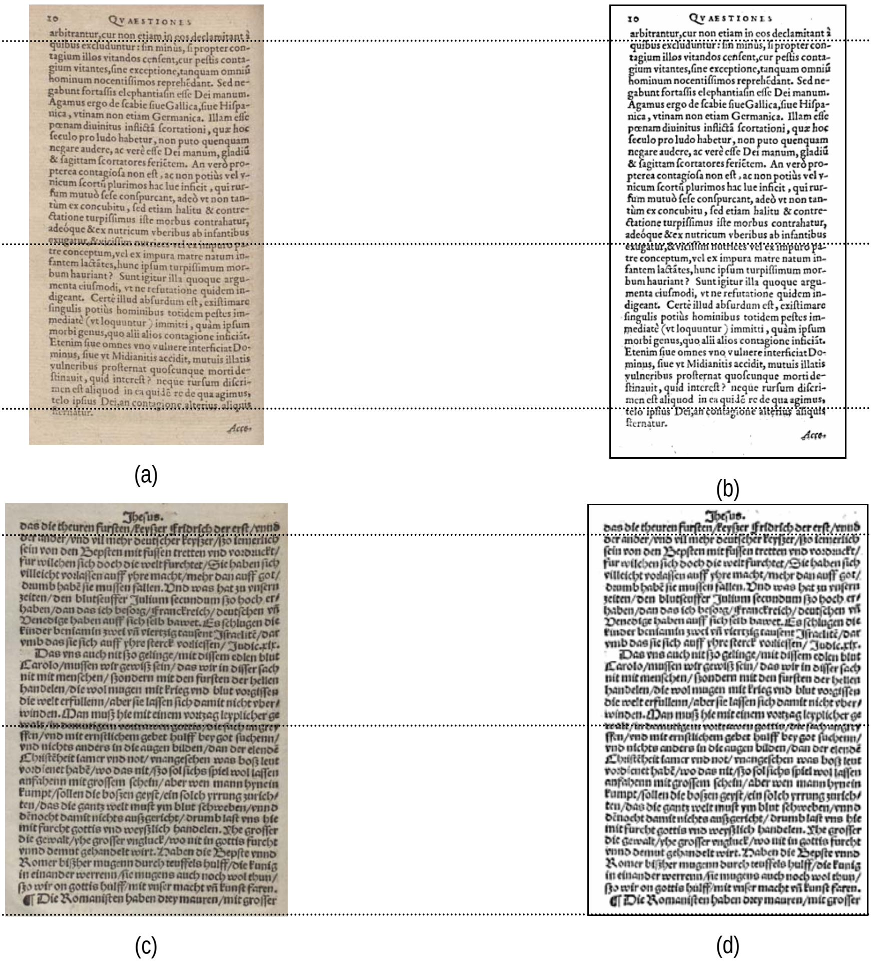
```
Page_Curl_Correction_v4 [0/1] [in] [out]
```

If the first parameter is 1 then only coarse rectification is applied otherwise coarse and fine rectification are applied. The new version incorporates a new method for the word baseline fitting and it also rectifies the distortion of individual words using baseline estimation. Finally, this version is able to process binary, grayscale or colour images. However, if the image is not binary only the coarse rectification will be applied.



(a) (b) (c)

Figure 30: Dewarping example; (a) original image; (b) coarse rectified image; (c) fine rectified image.



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

Figure 31: Page curl correction examples; (a),(c) original images; (b),(d) rectified images.

3.5. Evaluation

In order to measure the performance of the developed dewarping technique for historical documents we used the evaluation methodology we have proposed in [26]. This methodology 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) [27] 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 which must be corrected by a rectification technique. 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*:

$$\boldsymbol{DW} = \frac{\sum_{j=1}^{N} DW_j}{N} \times 100\% \qquad (4)$$

where DW_j is the measure which reflects the performance of the rectification technique with respect to the j^{th} selected text line and it is defined as follows:

$$DW_j = \begin{cases} 1 - \frac{Ar_j^{'}}{Ar_j}, & if \ \frac{Ar_j^{'}}{Ar_j} < 1 \\ 0 \quad , & otherwise \end{cases} \qquad (5)$$

where Ar_j and $Ar_j^{'}$ represent the integral of each cubic polynomial curve in distorted and rectified document images, over an interval delimited by the curve endpoints. Therefore, DW_j equals to one when the j^{th} 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 equal to or worse than the original image. Therefore, *DW* ranges in the interval [0,…,100] and the higher the value of *DW*, the better is the performance of the rectification technique.

For our experiments, we used 420 randomly selected images of historical documents (see Figure 33) and compared our results with those of the commercial product BookRestorer [29] (see Figure 34 and Figure 35). 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. The overall results are represented in the graph of Figure 32. As it can be observed, the proposed rectification method outperforms all the others methods. More specifically, it is recorded that it performs 7% better than BookRestorer.

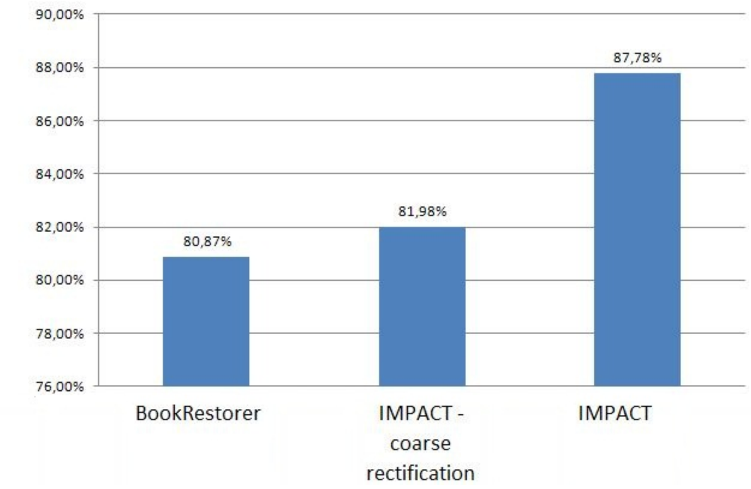
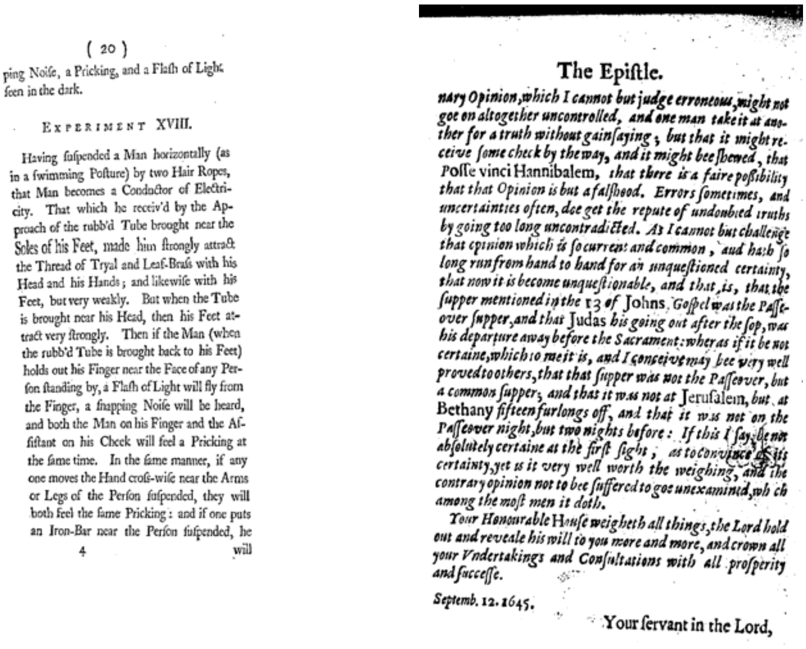


Figure 32: Evaluation results.



(20)

ping Noiſe, a Pricking, and a Flaſh of Light ſeen in the dark.

EXPERIMENT XVIII.

Having ſuſpended a Man horizontally (as in a ſwimming Poſture) by two Hair Ropes, that Man becomes a Conductor of Electricity. That which he receiv'd by the Approach of the rubb'd Tube brought near the Soles of his Feet, made him ſtrongly attract the Thread of Tryal and Leaf-Braſs with his Head and his Hands; and likewiſe with his Feet, but very weakly. But when the Tube is brought near his Head, then his Feet attract very ſtrongly. Then if the Man (when the rubb'd Tube is brought back to his Feet) holds out his Finger near the Face of any Perſon ſtanding by, a Flaſh of Light will fly from the Finger, a ſnapping Noiſe will be heard, and both the Man on his Finger and the Aſſiſtant on his Cheek will feel a Pricking at the ſame time. In the ſame manner, if any one moves the Hand croſs-wiſe near the Arms or Legs of the Perſon ſuſpended, they will both feel the ſame Pricking: and if one puts an Iron-Bar near the Perſon ſuſpended, he

4 will

The Epiſtle.

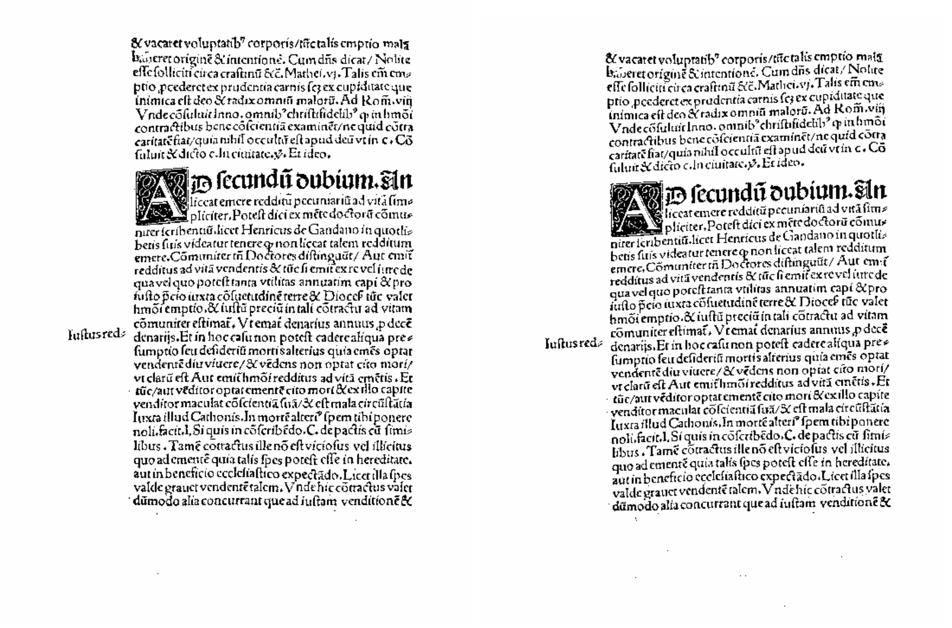
nary Opinion, which I cannot but judge erroneous, might not goe on altogether uncontrolled, and one man take it at another for a truth without gainſaying; but that it might receive ſome check by the way, and it might bee ſhewed, that Poſſe vinci Hannibalem, *that there is a faire poſſibility that that Opinion is but a falſhood. Errors ſometimes, and uncertainties often, doe get the repute of undoubted truths by going too long uncontradicted. As I cannot but challenge that opinion which is ſo current and common, and hath ſo long run from hand to hand for an unqueſtioned certainty, that now it is become unqueſtionable, and that is, that the ſupper mentioned in the* 13 *of* Johns *Goſpel was the Paſſeover ſupper, and that* Judas *his going out after the ſop, was his departure away before the Sacrament: wheras if it be not certaine, which to me it is, and I conceive may bee very well proved to others, that that ſupper was not the Paſſeover, but a common ſupper; and that it was not at* Jeruſalem, *but at* Bethany *fifteen furlongs off, and that it was not on the Paſſeover night, but two nights before: If this I ſay be not abſolutely certaine at the firſt ſight, as to convince of its certainty, yet is it very well worth the weighing, and the contrary opinion not to bee ſuffered to goe unexamined, wh ch among the moſt men it doth.*

Your Honourable Houſe *weigheth all things, the Lord hold out and reveale his will to you more and more, and crown all your Vndertakings and Conſultations with all proſperity and ſucceſſe.*

Septemb. 12. 1645.

Your ſervant in the Lord,

Figure 33: Example image of the evaluation set.



& vacaret voluptatib⁹ corporis/tūc talis emptio malā haberet originē & intentionē. Cum dn̄s dicat/ Nolite eſſe ſolliciti circa craſtinū &c̄. Mathei.vj. Talis em̄ emptio ꝓcederet ex prudentia carnis ſcꝫ ex cupiditate que inimica eſt deo & radix omniū malorū. Ad Rom̄.viij Vnde cōſuluit Inno. omnib⁹ chriſtifidelib⁹ ꝙ in hmōi contractibus bene cōſcientiā examinēt/ne quid cōtra caritatē fiat/quia nihil occultū eſt apud deū vt in c. Cōſuluit & dicto c. In ciuitate.ꝟ. Et ideo.

AD ſecundū dubium. An liceat emere redditū pecuniariū ad vitā ſimpliciter. Poteſt dici ex mēte doctorū cōmuniter ſcribentiū. licet Henricus de Gandano in quotlibetis ſuis videatur tenere ꝙ non liceat talem redditum emere. Cōmuniter tn̄ Doctores diſtinguūt/ Aut emit redditus ad vitā vendentis & tūc ſi emit ex re vel iure de qua vel quo poteſt tanta vtilitas annuatim capi & pro iuſto ꝓcio iuxta cōſuetudinē terre & Dioceſ tūc valet hmōi emptio. & iuſtū preciū in tali cōtractu ad vitam cōmuniter eſtimat̄. Vt emat̄ denarius annuus ꝓ decē denarijs. Et in hoc caſu non poteſt cadere aliqua preſumptio ſeu deſideriū mortis alterius quia emēs optat vendentē diu viuere/ & vēdens non optat cito mori/ vt clarū eſt Aut emit hmōi redditus ad vitā emētis. Et tūc/aut vēditor optat ementē cito mori & ex illo capite venditor maculat cōſcientiā ſuā/ & eſt mala circūſtātia Iuxta illud Cathonis. In mortē alteri⁹ ſpem tibi ponere noli. facit. l. Si quis in cōſcribēdo. C. de pactis cū ſimilibus. Tamē cōtractus ille nō eſt vicioſus vel illicitus quo ad ementē quia talis ſpes poteſt eſſe in hereditate. aut in beneficio eccleſiaſtico expectādo. Licet illa ſpes valde grauet vendentē talem. Vnde hic cōtractus valet dūmodo alia concurrant que ad iuſtam venditionē &

Iuſtus red-

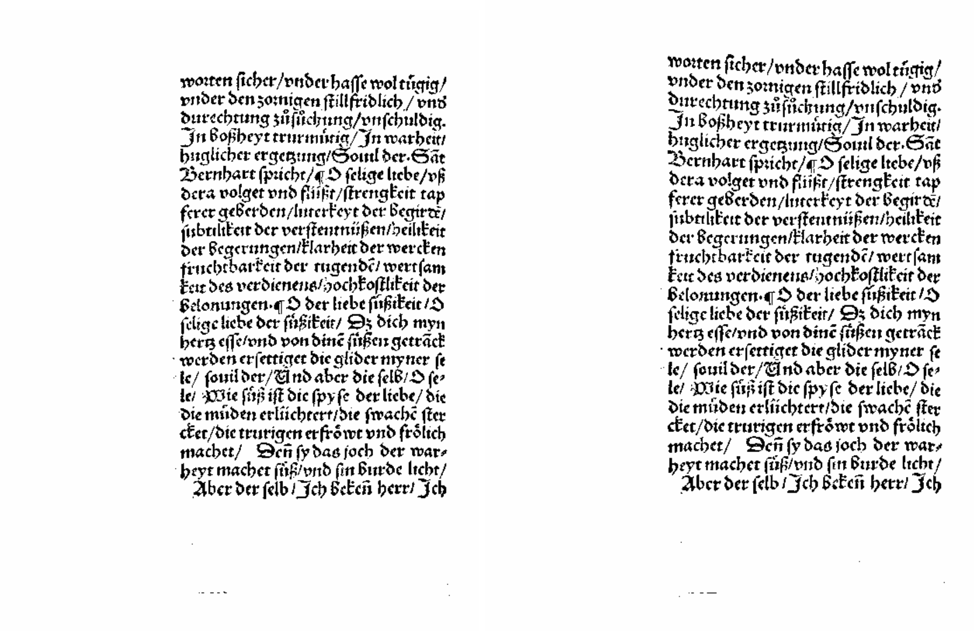
& vacaret voluptatib⁹ corporis/tūc talis emptio malā haberet originē & intentionē. Cum dn̄s dicat/ Nolite eſſe ſolliciti circa craſtinū &c̄. Mathei.vj. Talis em̄ emptio ꝓcederet ex prudentia carnis ſcꝫ ex cupiditate que inimica eſt deo & radix omniū malorū. Ad Rom̄.viij Vnde cōſuluit Inno. omnib⁹ chriſtifidelib⁹ ꝙ in hmōi contractibus bene cōſcientiā examinēt/ne quid cōtra caritatē fiat/quia nihil occultū eſt apud deū vt in c. Cōſuluit & dicto c. In ciuitate.ꝟ. Et ideo.

AD ſecundū dubium. An liceat emere redditū pecuniariū ad vitā ſimpliciter. Poteſt dici ex mēte doctorū cōmuniter ſcribentiū. licet Henricus de Gandano in quotlibetis ſuis videatur tenere ꝙ non liceat talem redditum emere. Cōmuniter tn̄ Doctores diſtinguūt/ Aut emit redditus ad vitā vendentis & tūc ſi emit ex re vel iure de qua vel quo poteſt tanta vtilitas annuatim capi & pro iuſto ꝓcio iuxta cōſuetudinē terre & Dioceſ tūc valet hmōi emptio. & iuſtū preciū in tali cōtractu ad vitam cōmuniter eſtimat̄. Vt emat̄ denarius annuus ꝓ decē denarijs. Et in hoc caſu non poteſt cadere aliqua preſumptio ſeu deſideriū mortis alterius quia emēs optat vendentē diu viuere/ & vēdens non optat cito mori/ vt clarū eſt Aut emit hmōi redditus ad vitā emētis. Et tūc/aut vēditor optat ementē cito mori & ex illo capite venditor maculat cōſcientiā ſuā/ & eſt mala circūſtātia Iuxta illud Cathonis. In mortē alteri⁹ ſpem tibi ponere noli. facit. l. Si quis in cōſcribēdo. C. de pactis cū ſimilibus. Tamē cōtractus ille nō eſt vicioſus vel illicitus quo ad ementē quia talis ſpes poteſt eſſe in hereditate. aut in beneficio eccleſiaſtico expectādo. Licet illa ſpes valde grauet vendentē talem. Vnde hic cōtractus valet dūmodo alia concurrant que ad iuſtam venditionē &

Iuſtus red-

(a) (b)

Figure 34: Rectification results using (a) the IMPACT dewarping toolkit, (b) the commercial product BookRestorer.



worten ſicher/vnder haſſe wol tügig/ vnder den zornigen ſtillfridlich / vnd durechtung zůſůchung/vnſchuldig. Jn boßheyt trurmütig/ Jn warheit/ hüglicher ergetzung/Souil der. Sāt Bernhart ſpricht/ ¶ O ſelige liebe/vß dera volget vnd flüßt/ſtrengkeit tapferer geberden/luterkeyt der begirdē/ ſubtilikeit der verſtentnüßen/heilikeit der begerungen/klarheit der wercken fruchtbarkeit der tugendē/ wertſamkeit des verdienens/hochkoſtlikeit der belonungen. ¶ O der liebe ſüßikeit / O ſelige liebe der ſüßikeit/ Dz dich myn hertz eſſe/vnd von dinē ſüßen getrāck werden erſettiget die glider myner ſele/ ſouil der/ Vnd aber die ſelb/ O ſele/ Wie ſüß iſt die ſpyſe der liebe/ die die müden erlüchtert/die ſwachē ſtercket/die trurigen erfröwt vnd frölich machet/ Deñ ſy das joch der warheyt machet ſüß/vnd ſin burde licht/ Aber der ſelb / Jch bekeñ herr/ Jch

worten ſicher/vnder haſſe wol tügig/ vnder den zornigen ſtillfridlich / vnd durechtung zůſůchung/vnſchuldig. Jn boßheyt trurmütig/ Jn warheit/ hüglicher ergetzung/Souil der. Sāt Bernhart ſpricht/ ¶ O ſelige liebe/vß dera volget vnd flüßt/ſtrengkeit tapferer geberden/luterkeyt der begirdē/ ſubtilikeit der verſtentnüßen/heilikeit der begerungen/klarheit der wercken fruchtbarkeit der tugendē/ wertſamkeit des verdienens/hochkoſtlikeit der belonungen. ¶ O der liebe ſüßikeit / O ſelige liebe der ſüßikeit/ Dz dich myn hertz eſſe/vnd von dinē ſüßen getrāck werden erſettiget die glider myner ſele/ ſouil der/ Vnd aber die ſelb/ O ſele/ Wie ſüß iſt die ſpyſe der liebe/ die die müden erlüchtert/die ſwachē ſtercket/die trurigen erfröwt vnd frölich machet/ Deñ ſy das joch der warheyt machet ſüß/vnd ſin burde licht/ Aber der ſelb / Jch bekeñ herr/ Jch

(a) (b)

Figure 35: Rectification results using (a) the IMPACT dewarping toolkit, (b) the commercial product BookRestorer.

Address the contribution towards project expectations. Tick the box according to which category your tool falls into:

☒	*Software tool is fit to be put into productive use and is supported by the necessary installation guides*
□	*Software tool can be made available in a productive environment with further development which is clearly defined*
□	*Software tool demonstrates potential functionality and is available in a publicly accessible environment*
□	*Report of findings of research available (for experimental tools only)*

3.6. License and IPR protection

This part of work is owned by the National Centre for Scientific Research "Demokritos", Greece (NCSR). After the project ends, individual licensing agreements for commercial and non-commercial use will be required.

3.7. References

[1] J. Liang, D. Doermann and H. Li, “Camera-based analysis of text and documents: a survey”, International Journal on Document Analysis and Recognition, vol. 7, no. 2-3, 2005, pp. 84–104.

[2] L. Zhang, Y. Zhang and C.L. Tan, “An Improved Physically-Based Method for Geometric Restoration of Distorted Document Images”, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 30, no. 4, 2008, pp. 728-734.

[3] A. Ulges, C.H. Lampert and T. Breuel, “Document capture using stereo vision”, ACM Symposium on Document Engineering, Milwaukee, Wisconsin, USA, 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”, 17th International Conference on Pattern Recognition, Cambridge, UK, 2004, pp. 482-485.

[5] M.S. Brown and W.B. Seales, “Image restoration of arbitrarily warped documents”, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 26, no. 10, 2004, pp. 1295-1306.

[6] H. Cao, X. Ding and C. Liu, “Rectifying the bound document image captured by the camera: A model based approach”, 7th International Conference on Document Analysis and Recognition, Scotland, 2003, pp. 71-75.

[7] J. Liang, D. DeMenthon and D. Doermann, “Geometric rectification of camera-captured document images”, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 30, no. 4, 2008, pp. 591-605.

[8] C.L. Tan, L. Zhang, Z. Zhang and T. Xia, “Restoring warped document images through 3D shape modeling”, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 28, no. 2, 2006, pp. 195-208.

[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 Recognition Journal, vol. 42, no. 11, 2009, pp. 2961-2978.

[10] O. Lavialle, X. Molines, F. Angella and P. Baylou, “Active Contours Network to Straighten Distorted Text Lines”, International Conference on Image Processing, Thessaloniki, Greece, 2001, pp. 748-751.

[11] C. Wu and G. Agam, “Document image De-warping for Text/Graphics recognition”, Joint IAPR International Workshop on Structural, Syntactic and Statistical Pattern Recognition, Windsor, Canada, 2002, pp. 348-357.

[12] L. Zhang and C.L. Tan, “Warped image restoration with applications to digital libraries”, 8th International Conference on Document Analysis and Recognition, Seoul, Korea, 2005, pp. 192-196.

[13] H. Ezaki, S. Uchida, A. Asano and H. Sakoe, “Dewarping of document image by global optimization”, 8th International Conference on Document Analysis and Recognition, Seoul, Korea, 2005, pp. 302-306.

[14] L. Mischke and W. Luther, “Document Image De-warping Based on Detection of Distorted Text Lines”, International Conference on Image Analysis and Processing, Cagliari, Italy, 2005, pp. 1068-1075.

[15] A. Ulges, C.H. Lampert and T.M. Breuel, “Document image dewarping using robust estimation of curled text lines”, 8th International Conference on Document Analysis and Recognition, Korea, 2005, pp. 1001-1005.

[16] 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 and Vision Computing, vol. 24, no. 8, 2006, pp. 837–848.

[17] D.C. Schneider, M. Block and R. Rojas, “Robust Document Warping with Interpolated Vector Fields”, 9th International Conference on Document Analysis and Recognition, Brazil, 2007, pp. 113-117.

[18] S.S. Bukhari, F. Shafait and T.M. Breuel, “Dewarping of document images using coupled-snakes”, Int. Workshop on Camera-Based Document Analysis and Recognition, Barcelona, Spain, 2009, pp. 34-41.

[19] B. Fu, M.Wu, R. Li, W. Li, Z. Xu and C. Yang, “A model-based book dewarping method using text line detection”, Int. Workshop on Camera-Based Document Analysis and Recognition, Brazil, 2007, pp. 63-70.

[20] Y. Zhang, C. Liu, X. Ding and Y. Zou, “Arbitrary warped document image restoration based on segmentation and Thin-Plate Splines”, 19th International Conference on Pattern Recognition, Florida, USA, 2008, pp. 1-4.

[21] M.S. Brown and Y.C. Tsoi, “Geometric and shading correction for images of printed materials using boundary”, IEEE Trans. on Image Processing, vol. 15, no. 6, 2006, pp. 1544-1554.

[22] A. Masalovitch and L. Mestetskiy, “Usage of continuous skeletal image representation for document images de-warping”, Int. Workshop on Camera-Based Document Analysis and Recognition, Brazil, 2007, pp. 45-53.

[23] i2S SA: http://www.i2s-bookscanner.com

[24] G. Farin, “Curves and Surfaces for Computer Aided Geometric Design: A practical guide”, 4th edition, Academic Press, 1996.

[25] S. Coons, “Surfaces for Computer Aided Design”, Technical report, Mass. Inst. Technol., Cambridge, 1968.

[26] N. Stamatopoulos, B. Gatos and I. Pratikakis, “A Methodology for Document Image Dewarping Techniques Performance Evaluation”, 10th International Conference on Document Analysis and Recognition, Barcelona, Spain, 2009, pp. 956-960.

[27] D.G. Lowe, “Distinctive image features from scale-invariant keypoints”, International Journal of Computer Vision, vol. 60, no. 2, 2004, pp. 91-110.