이학석사 학위논문

Document Image Dewarping based on the Document Boundary and 3D Reconstruction

(문서 경계와 3차원 재구성에 기반한 문서 이미지 평판화)

2022년 8월

서울대학교 대학원

수리과학부

전명재

Document Image Dewarping based on the Document Boundary and 3D Reconstruction

(문서 경계와 3차원 재구성에 기반한 문서 이미지 평판화)

지도교수 현동훈

이 논문을 이학석사 학위논문으로 제출함

2022년 4월

서울대학교 대학원

수리과학부

전명재

전명재의 이학석사 학위논문을 인준함

2022년 6월

위 원 장 천 정 희 (인)

부 위 원 장 현 동 훈 (인)

위 원 국 웅 (인)

Document Image Dewarping based on the Document Boundary and 3D Reconstruction

A dissertation
submitted in partial fulfillment
of the requirements for the degree of
Master of Science
to the faculty of the Graduate School of
Seoul National University

by

Myeong Jae Jeon

Dissertation Director : Professor David Donghoon Hyeon

Department of Mathematical Sciences
Seoul National University

August 2022

Abstract

In recent days, most of the scanned images are obtained from mobile devices such as cameras, smartphones, and tablets rather than traditional flatbed scanners. Contrary to the scanning process of the traditional scanners, capturing process of mobile devices might be accompanied by distortions in various forms such as perspective distortion, fold distortion, and page curls. In this thesis, we propose robust dewarping methods which correct such distortions based on the document boundary and 3D reconstruction. In the first method, we construct a curvilinear grid on the document image using the document boundary and reconstruct the document surface in the three dimensional space. Then we rectify the image using a family of local homographies computed from the reconstructed document surface. Although some of the steps of the proposed method have been proposed separately in other research, our approach exploited and combined their advantages to propose a robust dewarping process in addition to improving the stability in the overall process. Moreover, we refined the process by correcting the distorted text region boundary and developed this process into an independent dewarping method which is concise, straight-forward, and robust while still producing a well-rectified document image.

Key words: Document Image, Dewarping, Document boundary, Grid, 3D reconstruction
Student Number: 2018-29556

Contents

Chapter 1

Introduction

Increase in the usage of mobile devices such as smartphones and tablet computers made it easy for people to scan documents with these hand-held computers rather than using traditional flatbed scanners. Scanning process of smartphones or tablet computers is different from that of traditional flatbed scanners in the sense that the procedure takes a document image captured by a camera as an input. An issue is that document images captured by a camera may have distortions which prevent readers from identifying the text clearly. For example, perspective distortions may occur during the image capturing process using cameras. Even if we minimize perspective distortions by capturing an image at a good position, page curls on the document papers may cause another type of distortions on the document image. For example, page curls naturally occur near the spine of the book when the book is open. Even singularities might occur on the documents when a capturer holds the paper during the capturing process. Hence, pre-processing of the input document image is necessary to remove possible distortions on the text image. Since this pre-process rectifies a warped document image into the unwarped one, the process is called a dewarping process. In addition to the increasing demand in the handy scanning process using mobile devices, the advance in the optical character recognition (OCR) techniques which convert text images into the machine-encoded text also requires high quality document images. In these aspects, the dewarping process plays a significant role in many different areas handling text in the images.

Wagdy et al., [25] classified dewarping methods into three classes: (1) *Text information-based techniques*, (2)*Shape recovery-based techniques*, and (3)*Deep learning-based techniques*. According to their classification, text information-based techniques exploit the features of the document such as text-lines, words, and characters. On the other hand, shape recovery-based techniques depend on recovering the shape of the document. Lastly, deep learning-based techniques are dewarping methods based on deep learning techniques. Most of the dewarping methods use text information because it is one of the most prominent features in the document. However, it is also a challenging task that it is being researched actively to detect text-lines in the document image. Moreover, some of the documents might not contain enough text characters to use if they have figures or pictures instead. Meanwhile, deep learning techniques require strong computing power and memories throughout the process. Therefore, our first proposed method tries to recover the shape of the document.

Proposed Method The fundamental idea of the proposed method is to map local patches of the document image plane to local patches of the dewarped image plane. More specifically, a curvilinear grid enclosing the text-region will be constructed on the document image and mapped to a rectilinear grid on another plane by sending each warped grid cell to a rectangular grid cell using 4-point algorithm. Local transformations of grid cells will correct distortions in each grid cell and the target plane will contain the dewarped document image.

The difficulty in the above process is constructing rectangular target grid cells of appropriate size since the document surface in the image is distorted so that we cannot figure out the correct metric information of the document surface in the real scene. Therefore, we should understand the geometry of capturing process of the document surface in the real world. The dewarped document image might be identified with the original text on the flattened paper. We introduce a coordinate system (u, v) on another plane containing the dewarped image. Since the uv-plane is identified with the flattened paper, we may regard the document surface which is obtained by deforming the flattened paper as a parametric surface in $\mathbb{R}^3$ (the real world) with coordi-

nate system (x, y, z) defined on the uv-plane. That is, the document surface is a surface with isometric parametrization. Then we can also interpret the document image as an image of the document surface under a camera projection (or equivalently, a perspective projection) onto the image plane with coordinate system (x, y) - we may identify the image plane with the plane $z = 1$ in $\mathbb{R}^3$. Then, we reconstruct a curvilinear grid on the document surface from the previously constructed grid on the image plane. The grid on the document surface can also be interpreted as the image of the rectilinear grid on the uv-plane under the parametrization of the document surface. Since the parametrization is isometric, we may use the ratio of the sides of the grid cells on the reconstructed document surface to estimate the ratio of the size of the rectangular grid cells on the dewarped image on the uv-plane.

The proposed dewarping method consists of the following process. We first detect the document boundary consisting of top/bottom curves and side lines which might be tilted due to the perspective distortion of the captured image. Boundary edges in the document image represent the four edges of the document paper. Then, we first remove the perspective distortion with a homography computed from the 4-point algorithm using the four corners of the document boundary. After removing the perspective distortion, we interpolate boundary curves to construct a curvilinear grid on the document image. Each grid cell is regarded as a projection of a local patch on the document surface so that the document surface in the scene is reconstructed using the cost function designed in [24]. From the reconstructed document surface, we construct a collection of homographies using 4-point algorithm which map four corners of each warped grid cell on the document image to the four corners of a rectangular grid cell on the target image plane based on the size of the grid cells on the reconstructed document surface. Under these local homographies, the distortion is removed, and we obtain the dewarped image.

However, we sometimes identified little distortions on the image after the above dewarping process although the text was recognizable. To redeem this, we apply a global transformation to the image which corrects the distorted text region boundary. Moreover, we developed this process into an independent dewarping method by applying it to the document boundary.

We applied a global transformation to the text image with perspective distortion removed that transforms top/bottom boundary curves to the straight horizontal lines. Surprisingly, this process also produced a dewarped image of good quality. Hence, we propose this concise and effective method as an independent dewarping algorithm which can be applied separately.
The main contribution of the thesis is:

- Proposal of concise and robust dewarping methods based on the document boundary and 3D reconstruction.

More specifically, we propose

(1) a robust and efficient dewarping method based on the document boundary and 3D reconstruction consisting of

   (a) Construction of a curvilinear grid structure on the document image using the document boundary.

   (b) Improvement in the stability of the 3D reconstruction algorithm proposed in [24] by data normalization.

   (c) Application of a family of local homographies in dewarping process to remove various types of distortions in different areas on the document image.

   (d) Application of a global transformation which correct distorted text region boundary to remove the remained distortions.

(2) a concise and effective dewarping method based on a global transformation straightening document boundary curves.

Similar ideas to each step of the proposed method (1) have been proposed before. For example, [26] also constructed a curvilinear grid using the document boundary and stretched this to the straight and rectilinear grid. As stated before, [24] suggested the idea of 3D reconstruction in the dewarping process and mapped each grid cell of the reconstructed grid to the dewarped one. However, proposed method combined these steps which were proposed in other research and suggested an efficient and robust dewarping method while improving stability. Moreover, we developed a concise but still robust dewarping method (2) using document boundary. These are the main contributions of the thesis.

Contents The remaining chapters of the thesis are arranged as follows. In chapter 2, we briefly review camera geometry and a 3D reconstruction problem. In chapter 3, we review previous works on various dewarping algorithms related to the proposed methods. In chapter 4, we introduce the proposed dewarping method based on the document boundary and 3D reconstruction, and analyze the overall process in detail. In chapter 5, we propose the dewarping method which is developed from the supplementary process to remove the distortions in the dewarping method described in 4. Finally, in chapter 6, we conclude the discussion with future directions.

Chapter 2

Review on Camera Geometry

In this chapter, a brief review on camera geometry is given. Mathematical description of a basic camera model and a 3D reconstruction problem are stated. The contents in this chapter is based on the materials in [10].

2.1 Basic Camera Model

Pinhole Camera The simplest camera model is a pinhole camera. Geometrically, a pinhole camera is a central projection of points in the Euclidean space $\mathbb{R}^3$ onto the image plane. It maps a point $\mathbf{X} \in \mathbb{R}^3$ to the point $\mathbf{x}$ on the image plane where $\mathbf{x}$ is the intersection of the image plane and the line joining $\mathbf{X}$ and the centre of projection. For simplicity, let the centre of projection be the origin $O = (0, 0, 0)^\top \in \mathbb{R}^3$ and the image plane be the plane $Z = f$ where f is the distance between the centre and the image plane. Then, the camera maps a point $\mathbf{X} = (X, Y, Z)^\top \in \mathbb{R}^3$ to the point $\mathbf{x} = (fX/Z, fY/Z, f)^\top \in \mathbb{R}^3$ and its (x, y)-coordinate on the image plane can is $(fX/Z, fY/Z)^\top$. Thus, we may regard a pinhole camera as a map from $\mathbb{R}^3$ to $\mathbb{R}^2$ defined by

$$(X, Y, Z)^\top \mapsto (fX/Z, fY/Z)^\top \tag{2.1.1}$$

The centre of projection is called the *camera centre* and f will be called the *focal length* of the camera.

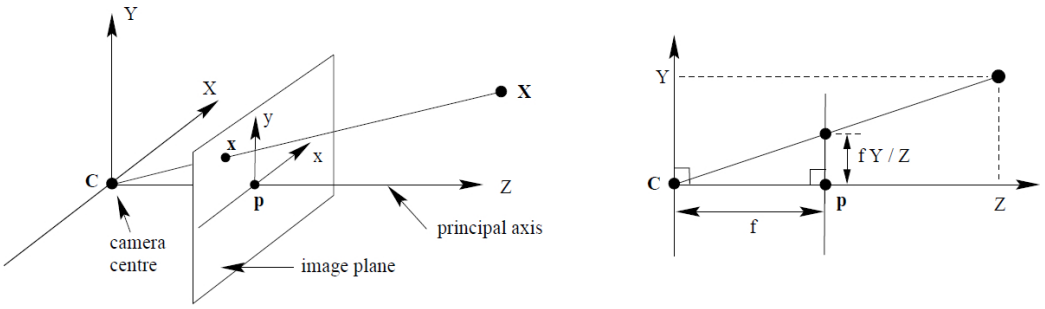


Figure 2.1: **Pinhole camera model**. The centre of projection is called the *camera centre*. The line from the camera centre perpendicular to the image plane is called the *principal axis*. The point where the principal axis meets the image plane is called the *principal point*. **C** is the camera centre and **p** the principal point. A point **x** on the image plane is the image of a scene point **X** under the camera projection.

Homogeneous Coordinates An observation is that given a point $\mathbf{X} \in \mathbb{R}^3$, a pinhole camera maps any point on the line joining the camera centre O and $\mathbf{X}$ to the same point on the image plane. Thus, we introduce a projective space and homogeneous coordinates. A *projective space* $\mathbb{P}^n$ *of dimension* n is a set of equivalence classes of points in $\mathbb{R}^{n+1}\backslash(0,\dots,0)^\top$ where $(X_0,\dots X_n)^\top \sim (X_0',\dots,X_n')^\top$ if and only if $(X_0,\dots,X_n)^\top = k(X_0,\dots,X_n)^\top$ for some nonzero $k \in \mathbb{R}$. For a point $\mathbf{X} \in \mathbb{P}^n$, any representative $(X_0,\dots,X_n)^\top$ of $\mathbf{X}$ is called a *homogeneous coordinate* of $\mathbf{X}$ and $(x_1,\dots,x_n)^\top$ is called an *inhomogeneous coordinate* of $\mathbf{X}$ if $(x_1,\dots,x_n,1)^\top$ is a homogeneous coordinate of $\mathbf{X}$.

With this concept, we identify the Euclidean coordinates of a world point and an image point with inhomogeneous coordinates of points in projective spaces. More specifically, (2.1.1) can be written in matrix multiplication

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX/Z \\ fY/Z \\ 1 \end{pmatrix} \sim \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{2.1.2}$$

Then, a pinhole camera can be interpreted as a map from $\mathbb{P}^3$ to $\mathbb{P}^2$ and

represented as a matrix

$$\begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.1.3}$$

up to constant multiplication - a homogeneous matrix. The matrix (2.1.3) is called a *camera matrix* of a camera. More generally, any camera represented by a homogeneous 3×4 matrix of rank 3 is called a *general projective camera* but we will only use a pinhole camera model.

2.2 3D Reconstruction Problem

An image captured by a camera is lack of information that we can perceive in the real world. Distortions occur during the capturing process. We cannot even see behind the objects in the image. However, a lot of information is delivered in the form of images or videos in recent days. Hence, retrieving the 3D scene from the captured images has become an important problem and it is called a *3D reconstruction* problem.

In general, we cannot reconstruct the scene from a single image due to the lack of information. For example, we cannot earn any information behind the objects in the image. Every point on the line from the camera centre is mapped to the same point on the image plane which makes it impossible to choose the real scene point. One may require additional information on the scene to proceed 3D reconstruction from a single image. Therefore, multiple images are used to obtain the 3D reconstruction of the scene in general. The process using multiple images in 3D reconstruction is called *stereo vision* which is applicable in many areas.

However, the given image is very special in our case. We only concerns document image which have its own characteristics such as text characters, text-lines and document boundary. We will see how to reconstruct the document surface (the document in the real space) from a single document image up to similarity relation based on such information. Although we cannot obtain the absolute metrics on the document surface, relative size of local regions will give us enough information to rectify the distorted document image

once we achieve 3D reconstruction of the document surface up to similarity.

Chapter 3

Related Works

In this chapter, we review previous works on dewarping algorithms related to the proposed method. Although the dewarping methods are divided into few sections based on their approaches, several approaches might be combined to provide a better result.

3.1 Dewarping Methods based on the Text-lines

Text-lines are the most commonly used features in various approaches since they are the most notable features in the text image. People can recognize horizontal text-lines in the document image with their eyes. However, it is a difficult task to detect text-lines correctly in the image processing step so that text-line detection algorithm has been researched thoroughly. For example, [15], [16] defined a state for each connected component which represent a text character. A state represents a local direction of a text-line which contains the connected component and local line-spacing near the connected component. A state for each connected component was estimated via optimization and they detected text-lines and text-blocks in the document image by joining the neighboring text characters based on their local directions.

Once detected and extracted in the document image, text-lines can play a significant role in dewarping process. For example, [13], [14] extracted text-

lines and text-blocks in the image and optimized cost functions which force the detected text-lines to be straight and aligned properly. On the other hand, [24] traced text-lines with a self-similarity measure and constructed a curvilinear grid on the document image and transformed this curvilinear grid to a rectilinear grid of appropriate size to remove the distortion due to bending of the document in the real world.

Although text-line information can be a powerful tool in the dewarping process, the problem is that there are some cases hard to use them properly. For example, some of the documents contain non-text objects such as figures, tables and even blank spaces. Mathematical texts contains mathematical equations which has different features from the standard text-lines in that they are relatively shorter than the usual text-lines and consist of numbers and symbols. At worst, the whole page might be full of pictures so that it may not contain a single text character. In such cases, we should find other information to use instead of text-lines. Document boundary might be one substitution.

3.2 Dewarping Methods based on the Document Boundary

We use the term document boundary as the four edges enclosing the text region. One can find such edges by detecting the four edges of the paper in the document image, or detecting the text region on the document image. The proposed method used the former one but one can define appropriate one according to their preference. The boundary edges are also notable features in the document image since we can detect them in the document image using their characteristics in the image. For example, page boundary edges often become the border of the illumination change in the image. Many works utilized such properties to detect boundary curves. For example, [11] firstly detected coarse corner points from the binary segmentation of book and background. Then exact corners, which give the book boundary, are detected from the extended boundaries extracted from the coarse corners. [26] reduced the resolution of the original image and used the borders of the white areas

in the binarized image. Canny edge detection method was also used on the small windows of the reduced image and the results were combined to detect boundary edges.

Once the boundary edges are detected, one can use them in dewarping process as in the text-line based dewarping methods. Most researches tried to transform curved boundary into the rectangular one to remove the distortion in the image due to the curvature of the document paper. To transform the boundary into the rectangle of appropriate size and ratio, [2], [3], [4] used Coons Patch interpolation which combines linear interpolation of the opposite boundary curves and bilinear interpolation of four corner points to geometrically correct the document image. [11] found the rectangle of appropriate size after reconstructing the document surface in the three dimensional space by finding the rulings of the document surface in the image using the side boundary lines and vanishing point. [22] transformed the curved boundary into the straight one by measuring the ratio of the lengths between the boundary curves and each pixel in the bounded region. [7] used the page boundary to find the vanishing point of two parallel vertical boundary lines and identified the images of the rulings of the document surface.

Dewarping methods based on the document boundary are useful in that they can be applied even when the document contains non-text objects or blank regions. However, there are also some cases that we can not detect document boundary when only the small portion of the document is captured in the image. Therefore, both methods using text lines and the document boundary might be used appropriately in appropriate cases. Our methods focus on the image that document boundary is detectable since it might be valid in most cases of the captured document image.

3.3 Dewarping Methods based on the Grid Construction

Dewarping methods based on grid construction on the image remove the distortions on the document image by transforming a curvilinear grid into a rectilinear one. Similar to the boundary-based dewarping methods, choosing

a rectilinear grid of appropriate size is the most essential procedure in the overall dewarping process. A grid on the image is constructed by choosing horizontal curves and vertical curves or lines whose intersections form the corner points of the grid. In most cases, a grid is constructed by using text-lines or top/bottom boundary edges as horizontal curves of the grid. Both methods may have their own advantages and disadvantages so that one can combine them appropriately to construct a proper grid.

[24] traced text-lines with a self-similarity measure and estimated the vertical directions by exploiting local stroke statistics in the text which allows to construct grid on the image. [7] used two opposite edges in an image of a paper if they are visible, and constructed a grid that its preimage consists of the square blocks of approximately the equal size. It is notable that the grid was formed by recognizing two or more parallel printed lines on the warped document image and extrapolating from there even when page boundaries are either unavailable or un-discernable. [8] even only used the image of a rectangular area on the text whose borders are parallel to the borders of the page and dewarp the image by applying transformations which flatten the grid structure bounded by the image of the rectangular region. [12] used deep learning networks DocUNet to detect the backward mapping of document image boundaries by regression and UNet to detect text lines by segmentation. Then they proposed a regularization module which takes the boundary constraint, text line constraint and the grid regularization term as optimization conditions to calculate a uniform forward map to dewarp the image.

3.4 Dewarping Methods based on the Document Surface Model in 3D Space

Many dewarping methods assume that the document forms a surface in the three dimensional space and use the geometry of the document surface and the image plane. Various surface models are used for assumption. The most simplest one is a cylindrical surface model so that many approaches [5], [8], [11], [13], [14] accepted the generalized cylindrical model for the surface.

Even more complicated but still simple surface model is the developable surface model [3], [17]. A developable surface is a smooth surface that can be flattened onto a plane without distortion. It is also plausible to accept developable surface model since the curved document in the image is originated from the flattened paper. Since developable surfaces in the three dimensional space $\mathbb{R}^3$ are ruled, one can try to find the images of the rulings in the document image and use their properties in dewarping methods. However, although the developable surface model can represent many document surfaces in the capturing process, there are still many cases that the model is not valid. For example, the document surface might have a singularity in the capturing process if they are wrinkled by non-predictable circumstances. Therefore, several approaches [22], [24] did not assume the specific model for the document surface.

Once the surface model is specified, each approach tried to exploit the geometric features such as spacing between text-lines on the document surface or arc-length between two points along the surface. From such features, several approaches try to optimize the cost function while others try to reconstruct the document surface in the dewarping process.

Chapter 4

Document Image Dewarping based on the Document Boundary and 3D Reconstruction

In this chapter, we describe the proposed dewarping method based on the document boundary and 3D reconstruction. As we have seen in chapter 3, diverse approaches are valid in dewarping process. The point is that we can utilize the various dewarping methods to exploit their advantages. Therefore, the proposed method in this chapter combines several existing methods in the dewarping process. First, we detect page boundary in the document image. From the detected document boundary, we construct a curvilinear grid on the image by interpolating top/bottom boundary curves and two side boundary lines. Then, we reconstruct the document surface in three dimensional space by optimizing the cost function proposed in [24] with the constructed grid. From the reconstructed surface, we compute the distance between the neighboring corner points of the grid on the reconstructed surface to find the rectilinear grid on a plane, with another coordinate system, which represents the flattened paper. Then, we map each pixel in the warped grid cells into the rectangular grid cells under the local homographies computed by the 4-point algorithm. Finally, we apply the global rectification to the image which cor-

rects the tilted text region boundary lines into the vertical or horizontal ones and remove the remaining distortions on the image. Moreover, we generalized the final process in the above method to the curved document boundary and developed into an independent dewarping method which also performs well. The second dewarping method will be described specifically in chapter 5.

Geometric Model As we have described in chapter 1, our objective is to map local patches of the document image to the local patches of the dewarped image. The coordinate system of the plane containing the dewarped image will be denoted by (u, v). The dewarped image on the uv-plane can be regarded as the unwarped text on the flattened paper. Hence, we want to map each warped local patch on the document image to the unwarped local patch of the uv-plane which represents the flattened paper. The problem is that we cannot estimate the real size of each local patch on the flattened paper from the document image due to distortions, so we introduce a document surface in the real world $\mathbb{R}^3$. The document surface is a curved paper in the real world $\mathbb{R}^3$ with the coordinate system (x, y, z) which can be considered as a parametric surface defined on the uv-plane. Intuitively, we may obtain the document surface by deforming the flattened paper in the scene. Therefore, we may assume that the parametrization of the document surface is isometric and estimate the size of local patches on the uv-plane by observing the document surface which will be reconstructed from the document image.

For reconstruction, we introduce a virtual camera whose centre and image plane are the origin $O = (0, 0, 0)^\top$ and $z = 1$, respectively. We regard the document image as the image of the document surface under this virtual camera projection. That is, the image point $\mathbf{x}$ of a scene point $\mathbf{X} = (X, Y, Z)^\top$ on the document surface is the intersection of the image plane $z = 1$ and the line joining the origin O and the scene point $\mathbf{X}$. So we identify the point $(x, y)^\top$ on the document image with the point $(x, y, 1)^\top$ in the real world. Then the document image becomes the image of a map

$$(X, Y, Z)^\top \mapsto (X/Z, Y/Z, 1)^\top.$$

Conversely, given the image point $\mathbf{x} = (x, y, 1)^\top$ on the image plane $z = 1$, the scene point of the image point $\mathbf{x}$ will lie on the back-projected line of the

image point $\mathbf{x}$. Then, the coordinate of the reconstructed scene point is of the form

$$(xz, yz, z)^\top = z(x, y, 1)^\top \in \mathbb{R}^3$$

and the task we have to do to reconstruct the document surface is determining the z-coordinate of the scene point for each image point. In general, there would be no clue to determine them from a single image but in our case, we can achieve reconstruction based on the remarkable features of the text image. Specifically, we use the document boundary to construct grid and attain its corner points G on the document image and reconstruct their scene points $\tilde{G}$ approximating the document surface by giving constraints on them.

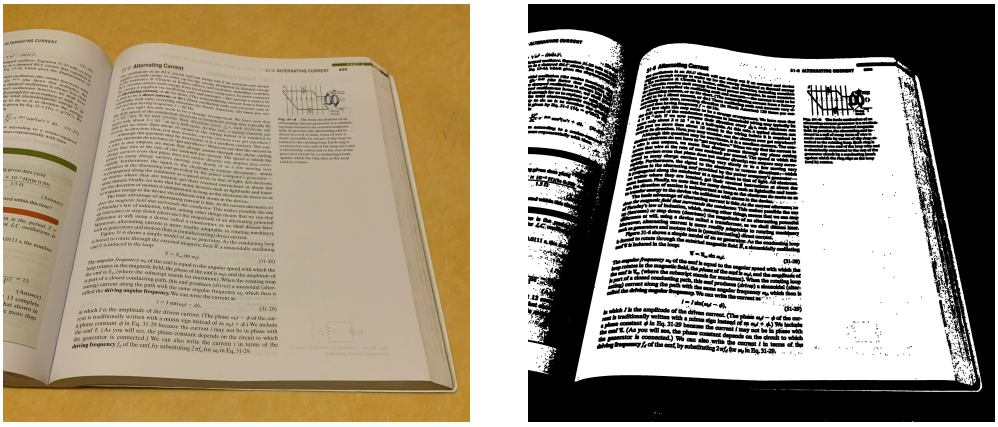
4.1 Input Document Image Processing

This section describes the preliminary image processing steps for the image dewarping. The image processing step consists of the binarization of the document image and document boundary detection from the document image. We also go through the first dewarping process which removes the perspective distortion based on the detected document boundary and the 4-point algorithm.

4.1.1 Binarization of the Input Document Image

The first step of the proposed method is binarization. Binarization is the process of converting the image into a binary (black and white) image. An input RGB document image is converted to a gray scale image which has 256 levels (0 to 255) of gray values and then binarized to the black and white binary image. Our goal in the binarization step is to obtain the clear binary image that discerns the text characters and the background regions.

Basically, binarization methods take a threshold to convert the gray scale image to the black and white image. More specifically, if the gray value of the pixel is greater than the threshold, then the pixel value is converted to 1 which represents the white pixel and if the gray value of the pixel is less than the threshold, the pixel value is converted to 0 which represents the



(a) (b)

Figure 4.1: Sample document images. (a)Input document image, (b) Binarized document image.

black pixel. One can choose a single threshold globally for the whole pixels. However, this approach may lose the local information of the image which may be important. Therefore, others may determine thresholds locally based on the regions for the accuracy of the binarization [20].

However, binarization is a difficult problem in most of the practical case. For example, in most of our cases, there exist differences in illumination on the document due to the relative positions of the document, the light source, and the camera during the image acquisition process which induce unpredictable shades on the document image. These shades may produce undesirable noise on the binarized image which makes it more difficult to detect text region. There also can be non-text regions on the documents such as tables or figures which make not only the binarization step but also the whole dewarping process more challenging and complicated. Moreover, including the document dewarping algorithm, most document analysis algorithms are built on taking advantage of the underlying binarized image data [21]. Due to these reasons, binarization of the image has been studied actively as an independent research topic [6], [18], [21], [23].

Since binarization algorithm is not our main interest, we adopt a binarization method produced by *MATLAB* [1], [19] and assume that we have a clear binary document image. However, any other state-of-the-art binariza-

tion algorithm would improve the proposed dewarping algorithm.

In the binarization step, we binarize the input document image and store the coordinate informations of the pixels which represent the text characters. We may assume that we can distinguish text characters and background clearly in our binarized document image for the rest of the dewarping process. From the binarized document image, we specify the coordinates of the pixels in all of the text characters in the document image.

4.1.2 Perspective Distortion Removal using the Document Boundary

After we specify the coordinates of the text characters in the image, we first remove the perspective distortion using document boundary. As indicated before, the document boundary means the four edges of the paper where the text is written. The most notable feature of the boundary edges is that we can observe the change in color intensity over the boundary edges. For example, when the captured document is located on the colored background, the boundary edges exactly become the boundary edges of the white region (which contains the text characters) in the image. Considering such characteristics, one can extract document boundary from the document image. In the proposed method, we detected the document boundary manually since it was not our main concern in the overall dewarping process. However, several document boundary detection methods have been proposed [11], [26] and better document boundary extraction method would improve the robustness of the dewarping process. Some document boundary detection methods are applied to two pages of the unfolded book while others are applied to a single page of the document. Our dewarping method will be applied to a single page since it is easy to extract a single page when we have the boundary of two pages in the image and in most cases, a single page will be captured in the capturing process.

Now we may assume that we have extracted document boundary from the image and denote the top/bottom edges by c_{top} and c_{bottom} and left/right edges by c_{left} and c_{right}. In most captured document image, top/bottom edges c_{top} and c_{bottom} are curved, while side edges c_{left} and c_{right} are straight. More-

over, side edges are tilted since the captured image is likely to have perspective distortion. Therefore, we first apply the transformation to the document image which removes the perspective distortion. From the four boundary edges of the text region, we can find four corners of the document boundary by finding intersections of the boundary edges. Let P_{TL}. P_{TR}, P_{BL}, and P_{BR} be four corners (top-left, top-right, bottom-left, and bottom-right, respectively). Our goal is to transform the four corners into four corners which form a rectangle. Therefore, we apply a homography computed from the 4-point algorithm to the image which maps four corners P_{TL}. P_{TR}, P_{BL}, P_{BR} to another set of four points which forms a rectangle of appropriate size.

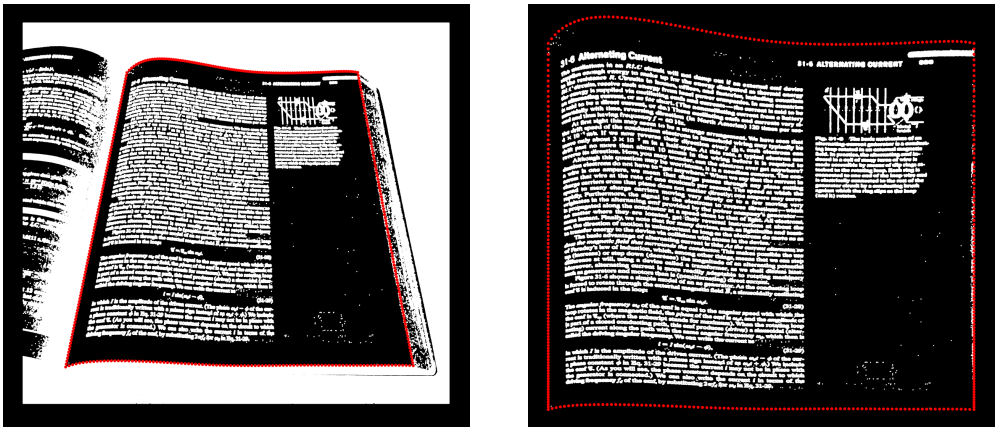
To determine the size of the target rectangle, we first fit top/bottom edges to the polynomial curves. The degree d of the fitted polynomials should be determined in the process. Several choices of d might be possible but we set $d = 9$ for the best result. Now we compute the lengths of the boundary edges in the initial document image and let

$$\begin{aligned} w &= \text{the length of the bottom boundary curve} \\ h &= \text{average of the lengths of two side boundary lines} \\ &= \frac{\overline{P_{\mathrm{TL}}P_{\mathrm{BL}}} + \overline{P_{\mathrm{TR}}P_{\mathrm{BR}}}}{2} \end{aligned}$$

From the computed length of boundary edges we set four points on the plane as below.

$$\begin{aligned} P'_{\mathrm{TL}} = (0,0), \quad P'_{\mathrm{TR}} = (w,0) \\ P'_{\mathrm{BL}} = (0,-h), \quad P'_{\mathrm{BR}} = (w,-h) \end{aligned}$$

Then, we compute the homography which maps each P_i to P'_i using the 4-point algorithm and apply this to the initial document image. This process removes the perspective distortion. It is worth to note that although we found the map converting the quadrilateral joining four corners of the boundary edges to the rectangle, the images of the top/bottom boundary curves are still curved in the second image since a homography maps a line to a line. So the document image acquired after the previous process will have four boundary edges consisting of two top/bottom boundary curves and two vertical side lines which are not tilted anymore as in figure 4.2b. As an abuse of notation,



(a) (b)

Figure 4.2: Sample input and output of the process in Section 4.1. (a) Document boundary is marked in red on the binarized document image, (b) Perspective distortion in the document image is removed by a homography and the transformed document boundary is marked in red.

the four corners and four boundary edges in the second image will still be denoted by P_{TL}. P_{TR}, P_{BL}, P_{BR} and c_{top}, c_{bottom}, c_{left}, c_{right} in the following section.

4.2 Grid Construction on the Document Image

In this section, we construct a curvilinear grid on the image with the document boundary obtained in the previous section. As a result of the process in the previous section, perspective distortion on the image is removed and the boundary edges of the document region consist of two curves (top/bottom edges c_{top}, c_{bottom}) and two vertical lines (side edges c_{left}, c_{right}). Moreover, the four corners P_{TL}. P_{TR}, P_{BL}, P_{BR} are also identified in the image.

We use boundary interpolation for grid construction. We first set the number m, n of grid cells in each column and row of the grid, respectively. Letting p_0 and p_m be polynomial equations, of degree $d = 9$, fitting the top/bottom boundary curves c_{top}, c_{bottom}, we interpolate p_0 and p_m to obtain

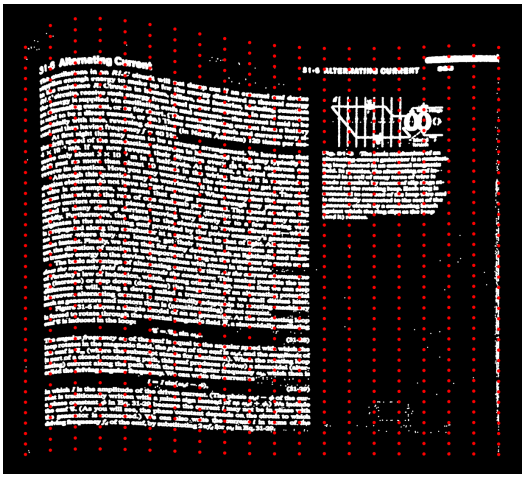


Figure 4.3: **Grid on the document image**. Grid corners are constructed by interpolating horizontal boundary edges and vertical side edges.

$m-1$ horizontal curves of the grid. Specifically, we set

$$p_i = \frac{m-i}{m} p_0 + \frac{i}{m} p_m, \quad i = 1, ..., m-1$$

We also interpolate both of the side boundary edges to specify corner points of the grid. Since our two side boundary edges are vertical lines $x = 0$ and $x = w$, we may choose $n+1$ regularly distributed points on each horizontal curve p_i and get corner points $\mathbf{p}_{ij}$ as below.

$$\begin{aligned} &\mathbf{p}_{ij} = (x_{ij}, p_i(x_{ij}), 1) = (x_{ij}, y_{ij}, 1), \\ &\text{where} \quad x_{ij} = \frac{j}{n} w, \quad i = 0, ..., m, \quad j = 0, ..., n \end{aligned}$$

Here we used the coordinate representation as a 3-tuple since we regard a pixel point in the image plane as a point in the projective plane $\mathbb{P}^2$ as well as a point on the image plane $z = 1$ embedded in $\mathbb{R}^3$. Then, $(m+1)(n+1)$ points form the set of corner points of the grid G.

$$G = \{\mathbf{p}_{ij}\}_{i,j=0}^{m,n}$$

4.3 3D Reconstruction of the Document Surface

In this section, we describe the reconstruction algorithm of the grid G in [24] which uses optimization of a cost function and improve the stability of the reconstruction process by data normalization.

4.3.1 Geometric Model

As described in the beginning of the chapter 4, we regard the input document image as the image of the document surface under the virtual camera whose centre and image plane are the origin $O = (0, 0, 0)^\top$ and $z = 1$. Then the document image is the image of the map

$$(X, Y, Z)^\top \mapsto (X/Z, Y/Z, 1)^\top$$

where each (X, Y, Z) is a point on the document surface. The scene point of the image point $(x, y, 1)^\top$ will lie on the back-projected line of the image point $(x, y, 1)^\top$. It is of the form

$$(xz, yz, z)^\top = z(x, y, 1)^\top \in \mathbb{R}^3$$

and the task in the reconstruction process of the document surface is determining the z-coordinate of the scene point for each image point. In general, there would be no clue to determine them from a single image. But in our case, we focus on grid point G constructed in section 4.2 and reconstruct their scene points $\tilde{G}$ by giving constraints on them.

A grid $\tilde{G}$ on the document surface is reconstructed up to similarity from the curvilinear grid G. The reconstruction is operated by optimizing the loss function which gives constraints on the desired reconstruction. The motivation for this process is interpreting grid cells of the grid G as a projection of the local patches of the document surface [24] under the camera matrix. Theoretically, the horizontal and vertical curves of the grid G may be back-projected to the parameter curves on the document surface and the points in $\tilde{G}$ correspond to the intersection points of the parameter curves. So the grid cells of $\tilde{G}$ will form the local patches of the document surface. If G is dense

enough, reconstructed $\tilde{G}$ would be sufficiently dense and approximate the document surface. Moreover, it is reasonable to assume that the grid cells of $\tilde{G}$ are parallelograms. Hence, the loss function in [24] was designed to force the grid cells in 3D space to be parallelograms and reconstruct the grid $\tilde{G}$ in 3D space.

Moreover, we regarded the document surface as an isometric parametric surface defined on the uv-plane which can be identified with the flattened paper. Hence, if we find the length of the horizontal curve of the reconstructed grid $\tilde{G}$, we may find the length of the horizontal text-lines on the uv-plane. In addition, by making a dense grid, the length of the curves on the document surface can be computed as the summation of the distance between the neighboring corner points of the grid without any integration. Thus, the distance between corner points will be used in the estimation of the rectangular grid cells on the uv-plane in section 4.4. To summarize, our goal in this section is stated as follows.

Objective Given a grid $G = \{\mathbf{p}_{ij}\}_{i,j=0}^{m,n}$ on the document image, find z_{ij}'s so that $\tilde{G} = \{z_{ij}\mathbf{p}_{ij}\}_{i,j=0}^{m,n} = \{\tilde{\mathbf{p}}_{ij}\}_{i,j=0}^{m,n}$ forms a grid on the document surface.

4.3.2 Normalization of the Grid Corners

The choice of a virtual camera in the previous step was quite arbitrary. A focal length of the virtual camera is set to be $f = 1$ without any information on the real camera which captured the input image. In fact, there are two problems we have to solve before applying the reconstruction algorithm in [24]:

(1) Choice of the image plane $Z = f$.

(2) Choice of the coordinate frame on the image plane and scaling of the image coordinates.

We have not mentioned about the answer of the problem (2) but f was arbitrarily chosen to be 1. We have tested several choices for the problem (1), (2) but different choices of them resulted in different reconstruction. Some results were not even close to the valid 3D reconstruction.

We solved these issues with data normalization. Given an input image $I = \{\mathbf{x}_1 = (x_1, y_1), \ldots, \mathbf{x}_N = (x_N, y_N)\}$ consisting of image points on the image plane, the scale and the coordinate origin can be arbitrarily chosen since it does not change the overall shape of the input image. However, the choice of the coordinates does affect the reconstruction result. Therefore, we normalize the input data to increase the accuracy of the algorithm and to make the algorithm unaffected by any choices of the scale and coordinate origin. The proposed method use normalization method called *isotropic scaling* which is introduced in [9], [10]. In our case, we use grid corners G as an input of the 3D reconstruction algorithm so that we normalize the points in G.

Isotropic scaling The normalization process consists of two steps. The first step is the translation of the coordinate origin. We translate the grid points to change their centroid into the origin. Then the coordinates are scaled so that their average (x, y, z)-coordinates have the similar magnitude. Since we let the grid points be on the plane $z = 1$, this is equivalent to scaling the points so that their average distance from the origin $(0, 0, 1)^\top$ is equal to $\sqrt{2}$. That is, setting average point to be $(1, 1, 1)^\top$. In summary, the normalization goes as follows:

1. The points are translated so that their centroid is at the origin: given $G = \{\mathbf{p}_{00}, \ldots, \mathbf{p}_{mn}\} = \{(x_{00}, y_{00}, 1)^\top, \ldots, (x_{mn}, y_{mn}, 1)^\top\}$, compute the centroid

$$(x_c, y_c, 1)^\top = (\frac{1}{mn}\sum_{i,j} x_{ij}, \frac{1}{mn}\sum_{i,j} y_{ij}, 1)^\top$$

   of G and replace $\mathbf{p}_{ij} = (x_{ij}, y_{ij}, 1)^\top$'s by $(x_{ij} - x_c, y_{ij} - y_c, 1)^\top$.

2. The points are then scaled so that the average distance from the origin is equal to $\sqrt{2}$: given $G = \{\mathbf{p}_{00}, \ldots, \mathbf{p}_{mn}\} = \{(x_{00}, y_{00}, 1)^\top, \ldots, (x_{mn}, y_{mn}, 1)^\top\}$, compute the average distance of points of G from the origin $(0, 0, 1)^\top$ of the image plane

$$d = \frac{1}{mn}\sum_{i,j} d_{ij} = \frac{1}{mn}\sum_{i,j}(x_{ij}^2 + y_{ij}^2)^{1/2}$$

and replace $\mathbf{p}_{ij} = (x_{ij}, y_{ij}, 1)^\top$'s by $(\frac{\sqrt{2}}{d}x_{ij}, \frac{\sqrt{2}}{d}y_{ij}, 1)^\top$.

Skipping each step in normalization process induced unexpected results when the reconstruction algorithm was applied. However, the normalization step allowed to obtain reasonable and stable 3D reconstruction of the grid corners representing the document surface.

4.3.3 3D Reconstruction of the Document Surface

Now we describe the 3D reconstruction algorithm in [24] which is applied in our approach. Many dewarping algorithms using 3D reconstruction of the document surface assumed the generalized cylindrical surface (GCS) model for the document surface [5], [8], [11], [13], [14]. It assumes that the document surface is a cylinder and exploits cylindrical properties in reconstruction step. This assumption lacks theoretical justification since most of the document surface would have more complex structure. More generalized surface model, developable surface model, was applied in some cases [3], [17], but this model is still distinguished from the general surface. On the other hand, the reconstruction algorithm in the proposed method is applicable to more general shape of document surfaces.

The key idea is regarding grid cells of G as the images of local patches of the document surface. If constructed grid G is dense enough, it is reasonable to assume that back-projection of grid curves (horizontal and vertical curves constructed in section 4.2) form parameter curves on the document surface so that back-projected grid cells are the local patches of the document surface in the shape of parallelograms. Thus, [24] designed a cost function which constrains the reconstructed grid cells to be parallelograms. The cost function is designed as follows.

Let $\mathbf{p}_{i+1,j}$, $\mathbf{p}_{i+1,j+1}$, $\mathbf{p}_{i,j+1}$ and $\mathbf{p}_{ij}$ be four corner points of the (i,j)-th grid cell of the grid G on the document image. Then the reconstructed points of $\mathbf{p}_{kl}$'s are of the form $\tilde{\mathbf{p}}_{kl} = z_{kl}\mathbf{p}_{kl}$ as described in section 4.3.1. Then the reconstructed grid cell with four corner points $\tilde{\mathbf{p}}_{i+1,j}$, $\tilde{\mathbf{p}}_{i+1,j+1}$, $\tilde{\mathbf{p}}_{i,j+1}$ and $\tilde{\mathbf{p}}_{ij}$ form a parallelogram if and only if they satisfy the equation

$$\tilde{\mathbf{p}}_{i+1,j} - \tilde{\mathbf{p}}_{i+1,j+1} + \tilde{\mathbf{p}}_{i,j+1} - \tilde{\mathbf{p}}_{ij} = 0 \tag{4.3.1}$$

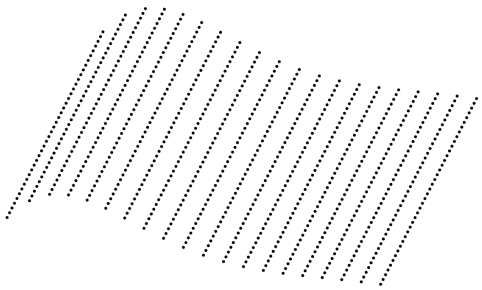


Figure 4.4: **Reconstructed document surface**. Reconstructed 3D grid corners approximate the document surface.

Since $\tilde{\mathbf{p}}_{kl} = z_{kl}\mathbf{p}_{kl}$, the following cost function is minimized:

$$L(\{z_{ij}\}_{i,j=0}^{m,n}) = \sum_{i,j=0}^{m-1,n-1} \|z_{i+1,j}\mathbf{p}_{i+1,j} - z_{i+1,j+1}\mathbf{p}_{i+1,j+1} + z_{i,j+1}\mathbf{p}_{i,j+1} - z_{ij}\mathbf{p}_{ij}\|^2 \tag{4.3.2}$$

To avoid the trivial solution $Z = (z_{00}, \ldots, z_{mn})^\top = \mathbf{0}^\top$, a global scale constraint $\|Z\| = 1$ is added. The cost function of this form under the given constraint can be minimized by linear least-squares method, in particular, singular value decomposition (SVD). It is worth to note that minimizing L is a linear problem so that it does not cost much time. Consequently, we can efficiently find Z minimizing the cost function L and obtain reconstructed grid $\tilde{G}$ on the document surface. Details of the minimization process finding optimal Z based on SVD is provided in appendix A.2.

4.4 Rectification of the Document Image under a Family of Local Homographies

In this section, we apply a family of homographies to the document image which remove the overall distortion in the image. More specifically, we com-

pute local homographies using the 4-point algorithm which map each grid cell in the distorted image to the corresponding grid cell of the rectilinear grid on the uv-plane. Finally, we can remove the distortion by transforming warped grid cells to the unwarped ones by these local homographies. We summarize the notations so far and describe the process in this section.

The grid G on the image plane and the reconstructed grid $\tilde{G}$ are denoted by

$$G = \{\mathbf{p}_{00}, \ldots, \mathbf{p}_{mn}\},\ \tilde{G} = \{\tilde{\mathbf{p}}_{00}, \ldots, \tilde{\mathbf{p}}_{mn}\}.$$

The (i,j)-th grid cell with corner points $\mathbf{p}_{i+1,j},\ \mathbf{p}_{i+1,j+1},\ \mathbf{p}_{i,j+1},\ \mathbf{p}_{ij}$ and the (i,j)-th reconstructed grid cell with corner points $\tilde{\mathbf{p}}_{i+1,j},\ \tilde{\mathbf{p}}_{i+1,j+1},\ \tilde{\mathbf{p}}_{i,j+1},\ \tilde{\mathbf{p}}_{ij}$ will be denoted by C_{ij} and $\tilde{C}_{ij}$, respectively. The grid G is transformed to the grid G' on the uv-plane which will be constructed as follows.

Horizontal Spacing of the Grid For each $i = 0, \ldots, m$ and $j = 0, \ldots, n-1$, compute

$$w_{ij} = d(\tilde{\mathbf{p}}_{ij}, \tilde{\mathbf{p}}_{i,j+1})$$

where $d(\mathbf{x}, \mathbf{y})$ is the Euclidean distance between two points $\mathbf{x}, \mathbf{y} \in \mathbb{R}^3$. Then the mean horizontal spacing w_j of the j-th column of the grid $\tilde{G}$ is computed:

$$w_j = \frac{1}{m+1} \sum_{i=0}^{m} w_{ij}, \quad j = 0, \ldots, n-1$$

Vertical Spacing of the Grid Similarly, the mean vertical spacing h_i of the i-th row of the grid $\tilde{G}$ is computed as below.

$$h_{ij} = d(\tilde{\mathbf{p}}_{ij}, \tilde{\mathbf{p}}_{i+1,j})$$
$$h_i = \frac{1}{n+1} \sum_{j=0}^{n} h_{ij}, \quad i = 0, \ldots, m-1, \quad j = 0, \ldots, n$$

Then we construct a rectilinear grid G' on the uv-plane with (i,j)-th corner point

$$\mathbf{p}'_{ij} = (\sum_{l=0}^{j-1} w_j, -\sum_{k=0}^{i-1} h_i)^\top, \quad i = 0, \ldots, m, \quad j = 0, \ldots, n.$$

whose x-coordinate is a sum of mean horizontal spacing of 0, ..., $(j-1)$-th columns and y-coordinate is a negative sum of mean vertical spacing of 0, ..., $(i-1)$-th rows. For convenience, we identify the point $(x', y')^\top$ on the uv-plane with the point $(x', y', 1)^\top$ as before. Then, a family of local homographies

$$H_{ij} : C_{ij} \to C'_{ij}, \quad i = 0, \ldots, m-1, \quad j = 0, \ldots, n-1$$

is computed from the 4-point algorithm. We take the four corner points of C_{ij} and C'_{ij} for 4-point correspondences. Explicitly, H_{ij} is a homography computed from the 4-point algorithm with point correspondences

$$\begin{aligned} \mathbf{p}_{ij} &\mapsto \mathbf{p}'_{ij} \\ \mathbf{p}_{i+1,j} &\mapsto \mathbf{p}'_{i+1,j} \\ \mathbf{p}_{i+1,j+1} &\mapsto \mathbf{p}'_{i+1,j+1} \\ \mathbf{p}_{i,j+1} &\mapsto \mathbf{p}'_{i,j+1}. \end{aligned}$$

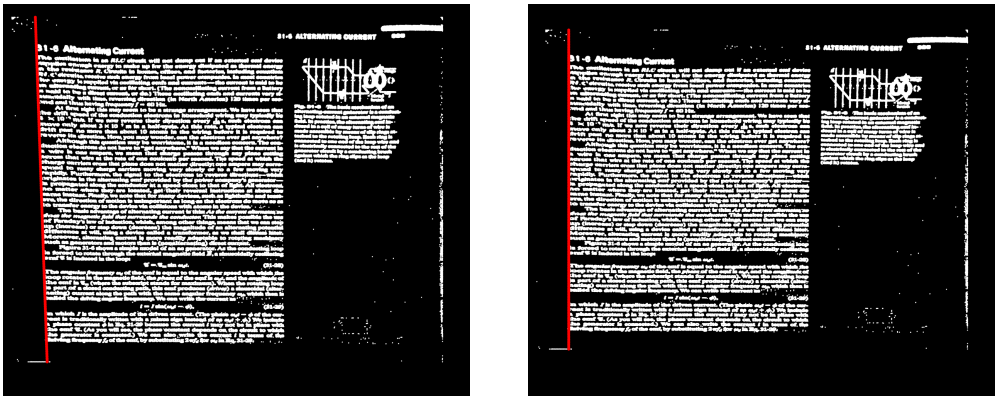
Then we map each point $\mathbf{x}_k = (x_k, y_k, 1)^\top$ in the boundary and the interior of the grid cell C_{ij} to the point $\mathbf{x}'_k = H_{ij}\mathbf{x}_k = (x'_k, y'_k, w'_k)^\top$ by H_{ij}.

$$\begin{aligned} &H_{ij} : C_{ij} \to C'_{ij} \\ &\qquad \mathbf{x}_k \mapsto \mathbf{x}'_k = H_{ij}\mathbf{x}_k = (x'_k, y'_k, w'_k)^\top \sim (\frac{x'_k}{w'_k}, \frac{y'_k}{w'_k}, 1)^\top \end{aligned}$$

Note that if an image point $\mathbf{x}$ lies on the boundary of two distinct grid cells, say C_{ij} and C_{kl}, the point will be mapped to the same point under two distinct homographies H_{ij} and H_{kl} since homographies are linear maps so that they preserve the line. Finally, we take inhomogeneous coordinate $(x'_k/w'_k, y'_k/w'_k)^\top$ of the image point to locate the point on the uv-plane. By collecting all of these points on the uv-plane, we obtain the rectified document image.

4.5 Global Rectification of the Document Image

In this section, we complement the rectification process by applying the global transformation on the image obtained in the previous section. We have ob-



(a) (b)

Figure 4.5: Dewarping process. (a) Document image rectified by a family of local homographies in section 4.4 with tilted left boundary, (b) Document image rectified by a supplementary global transformation in section 4.5 with corrected left boudnary.

served that little distortion is remained during the experiments. To be specific, we detected that text region boundary was not an intended rectangle. Text region boundary here means the smallest quadrilateral enclosing the text region, not the page boundary. For example, the left side of the boundary edge is tilted in 4.5a, not vertical. One might apply the 4-point algorithm again to map the four corners of the text region boundary to the four corners of the rectangle, but we suggest another method which can be devloped to an independent dewarping method in chapter 5.

To find the boundary edges enclosing the text region, we first find the leftmost point in each grid cell of the first column of the grid and find the line fitting these points. Similarly, we can find four boundary lines enclosing the text region, say l_{top}, l_{bottom}, l_{left}, and l_{right} for top, bottom, left and right boundary line, respectively. We observed some of them are not horizontal nor vertical, but in the desired image, top/bottom lines must be horizontal and left/right lines must be vertical. Therefore, we apply a transformation on the image which makes the tilted boundary line straight. For example, assume that the left boundary line is tilted. We first translate the image so that the whole image is located on the left side of the y-axis (We identify the uv-plane

with the xy-plane in this section). Specifically, we apply the translation

$$(x, y) \mapsto (x - x_{\max}, y)$$

to each point (x, y) in the image where $x_{\max}$ is the maximum x-coordinate of the points in the image. After translation, let the equation of the left boundary line be

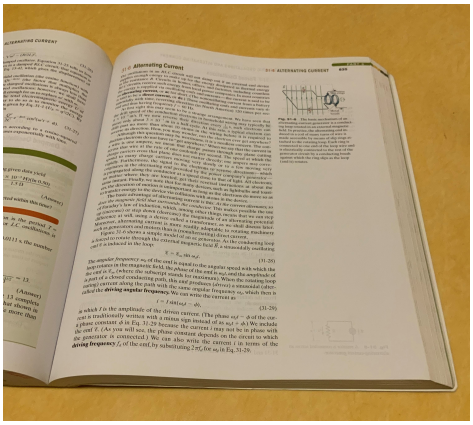
$$x = ay + b, \quad a, b \in \mathbb{R}$$

Then we apply the transformation

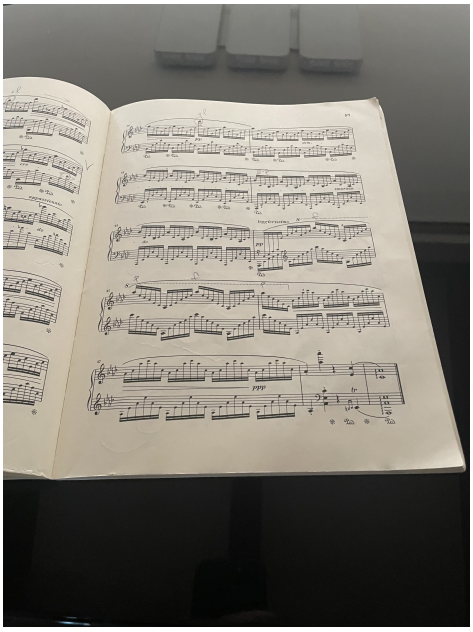
$$(x, y) \mapsto (\frac{x_{\min} x}{ay + b}, y)$$

to each point (x, y) in the translated image where $x_{\min}$ is the minimum x-coordinate of the points in the translated image. Note that this transformation maps each point (x, y) on the left boundary line to $(x_{\min}, y)$ so that the tilted boundary line is transformed to the vertical line $x = x_{\min}$. Applying the similar procedure to each tilted boundary line, we can transform the text region boundary into the rectangle and obtain the final image.

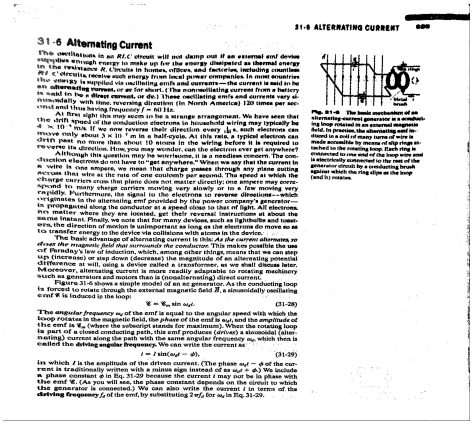
Although the above process was designed as an additional step of the dewarping process at first, we verified that the image can be well-dewarped if it is applied to the document boundary including curved edges detected in section 4.1. Thus, we propose an independent dewarping method in chapter 5 based on this section.



(a)



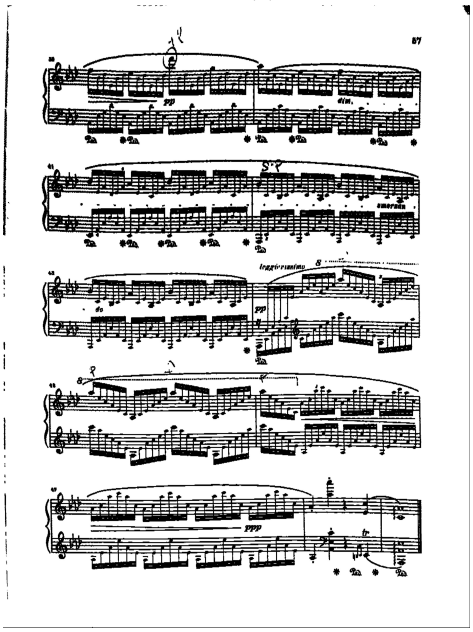
(b)



31-6 ALTERNATING CURRENT

31-6 Alternating Current

(c)



(d)

Figure 4.6: **Dewarping results**. Document images rectified by the proposed method in chapter 4. (a) Input text image, (b) Input score image, (c) Dewarped text image, (d) Dewarped score image.

Chapter 5

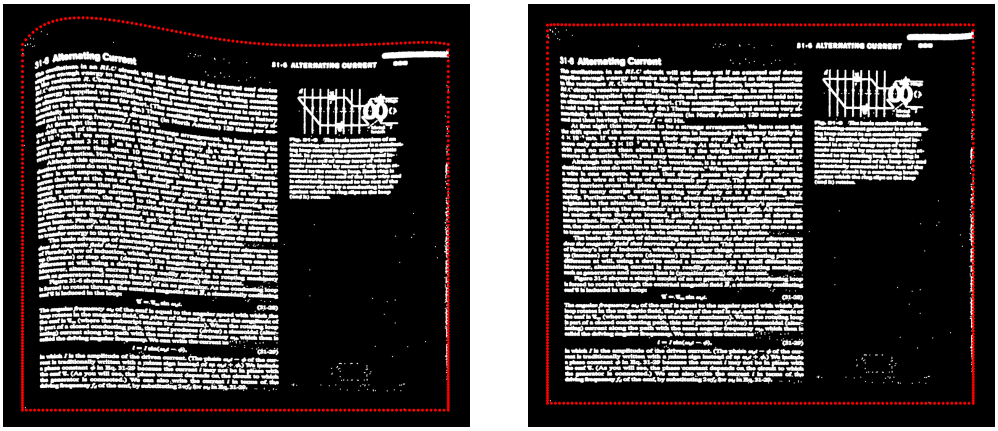
Document Image Dewarping by Straightening Document Boundary Curves

In this section, we describe the dewarping method based on the supplementray rectification process proposed in section 4.5. The overall process is very concise, robust and result in well-dewarped images.

Perspective Distortion Removal Firstly, we operate the same process described in section 4.1. That is, we detect document boundary and remove the perspective distortion in the document image using the homography computed from the 4-point algorithm and four corners of the document boundary. As a result, we have document boundary consisting of two curves (top/bottom edges) and two vertical lines (left/right edges).

Boundary Curve Straightening Now we apply the transformations similar to the ones described in section 4.5 to the image which straighten top/bottom boundary curves. We first translate the image so that the image is located below the x-axis. This can be done by applying the translation

$$(x, y) \mapsto (x, y - y_{\max})$$



(a) (b)

Figure 5.1: Dewarping process. (a) Document image rectified by a global transformation straightening the bottom boundary curve, (b) Final document image rectified by a global transformation straightening the top boundary curve. Document boundary in each process is marked in red.

to each point (x, y) in the image where $y_{\max}$ is the maximum y-coordinate of the points in the image. Letting $y = p_{\text{bottom}}(x)$ be the degree $d = 9$ polynomial equation fitting the bottom boundary curve, we apply the transformation

$$(x, y) \mapsto (x, \frac{y_{\min} y}{p_{\text{bottom}}(x)})$$

to each point (x, y) in the translated image where $y_{\min}$ is the minimum y-coordinate of the points in the translated image. Note that if (x, y) is a point on the bottom boundary curve, it is mapped to the point $(x, y_{\min})$ so that the bottom boundary curve is mapped to the horizontal line $y = y_{\min}$.

After the above transformations, the top boundary edge is still remained curved while the side boundary edges are remained as the same vertical lines. Thus, we apply the similar transformations to the image to straighten the top boundary curve.

Similar to the previous case, we translate the image so that it is located above the x-axis. It is done by the translation

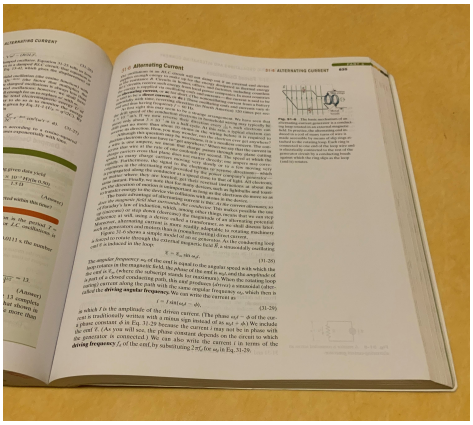
$$(x, y) \mapsto (x, y - y_{\min})$$

of point (x, y) in the image where $y_{\min}$ is the minimum y-coordinate of the points in the image. Letting $y = p_{\text{top}}(x)$ be the degree $d = 9$ polynomial equation fitting the top boundary curve, we apply the transformation

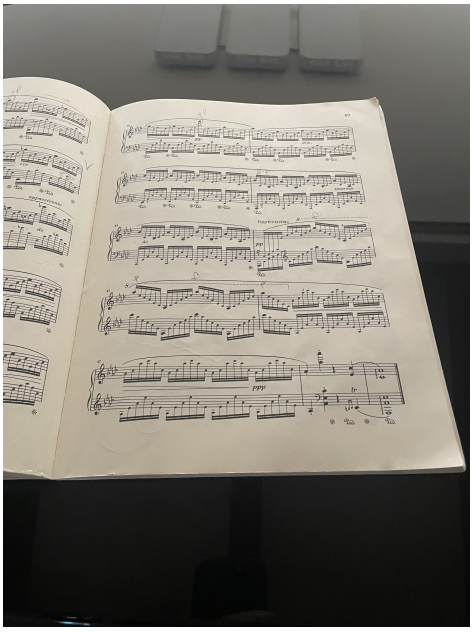
$$(x, y) \mapsto (x, \frac{y_{\max} y}{p_{\text{top}}(x)})$$

to each point (x, y) in the translated image where $y_{\max}$ is the maximum y-coordinate of the points in the translated image. Note that if (x, y) is a point on the top boundary curve, it is mapped to the point $(x, y_{\max})$ so that the top boundary curve is mapped to the horizontal line $y = y_{\max}$.

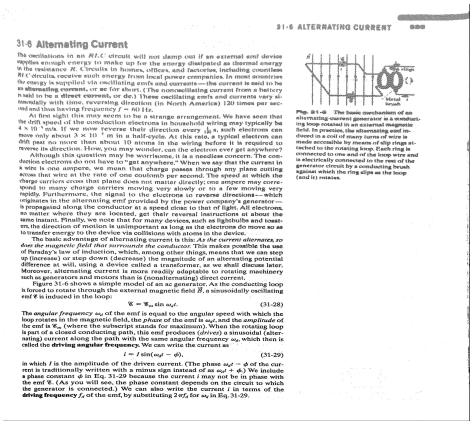
As a result, we obtain the rectified image with rectangular text region boundary whose four sides consisting of vertical and horizontal lines. Although the whole process is concise, we could get a meaningful result in removing distortion.



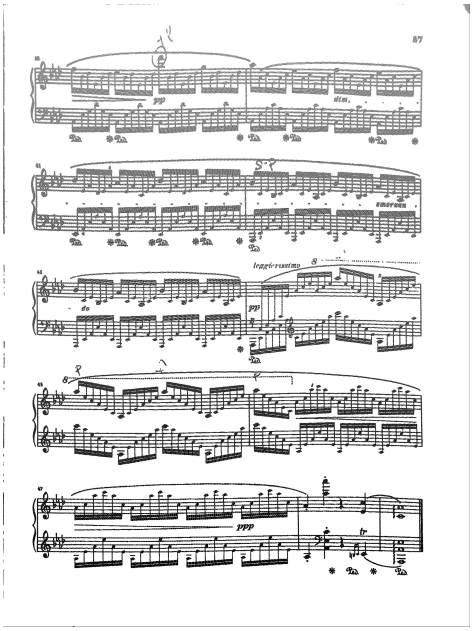
(a)



(b)



(c)



(d)

Figure 5.2: **Dewarping results**. Document images rectified by the proposed method in chapter 5. (a) Input text image, (b) Input score image, (c) Dewarped text image, (d) Dewarped score image.

Chapter 6

Conclusion

In this dissertation, two robust document image dewarping methods have been proposed. One was to utilize and refine previous works to produce a better result. We first detected document boundary from the image and removed the perspective distortion by applying the homography computed from the 4-point algorithm. Then we constructed a curvilinear grid on the image by interpolating the boundary edges which are regarded as the image of the local patches of the document surface. From the constructed grid, we reconstructed 3D grid corners approximating the document surface after normalizing the constructed grid corners on the image. Then we applied a family of local homographies to the image which was computed from the distances between neighboring grid corners on the document surface. Finally, we corrected distorted text region boundary edges to remove the remaining distortions. Moreover, we developed this supplementary process into another dewarping method which is concise, robust, and produces good result.

The overall dewarping process inevitably depends on the result of the image processing step. We expect that the better image processing method would improve the proposed method.

Appendix A

A.1 4-point Algorithm

A *homography* or a *projective transformation* of projective spaces $\mathbb{P}^n$ is an isomorphism of projective spaces induced by an isomorphism of vector spaces $\mathbb{R}^{n+1}$. Equivalently, it is a linear transformation on homogeneous $(n+1)$-vectors represented by a non-singular $(n+1)\times(n+1)$ matrix. In particular, a homography between projective planes $\mathbb{P}^2$ is called a planar homography. In this section, we describe the *4-point algorithm* [10] finding the planar homography H used in the proposed dewarping method.

Objective Given $n = 4$ 2D to 2D homogeneous point correspondences $\{\mathbf{x}_i \mapsto \mathbf{x}'_i\}$, determine the 3×3 homography matrix H such that $\mathbf{x}'_i = H\mathbf{x}_i$.

Using the *4-point algorithm*, we can achieve the above objective. Let the coordinates of the corresponding points be

$$\mathbf{x}_i = (x_i, y_i, w_i)^\top, \quad \mathbf{x}'_i = (x'_i, y'_i, w'_i)^\top. \tag{A.1.1}$$

Since a point in projective space is determined up to non-zero scalar multiplication, $\mathbf{x}'_i$ and $H\mathbf{x}_i$ do not have to be equal, but parallel 3-dimensional vectors. Therefore, the equation is equivalent to

$$\mathbf{x}'_i \times H\mathbf{x}_i = \mathbf{0}. \tag{A.1.2}$$

APPENDIX A.

Denote j-th row of the homography matrix H by $\mathbf{h}^{j\top}$. Then we may write

$$H\mathbf{x}_i = \begin{pmatrix} \mathbf{h}^{1\top}\mathbf{x}_i \\ \mathbf{h}^{2\top}\mathbf{x}_i \\ \mathbf{h}^{3\top}\mathbf{x}_i \end{pmatrix}$$

and equation (A.1.2) can be written explicitly in terms of coordinates (A.1.1) as

$$\mathbf{x}'_i \times H\mathbf{x}_i = \begin{pmatrix} y'_i\mathbf{h}^{3\top}\mathbf{x}_i - w'_i\mathbf{h}^{2\top}\mathbf{x}_i \\ w'_i\mathbf{h}^{1\top}\mathbf{x}_i - x'_i\mathbf{h}^{3\top}\mathbf{x}_i \\ x'_i\mathbf{h}^{2\top}\mathbf{x}_i - y'_i\mathbf{h}^{1\top}\mathbf{x}_i \end{pmatrix} = \mathbf{0}.$$

Since $\mathbf{h}^{j\top}\mathbf{x}_i = \mathbf{x}_i^\top\mathbf{h}^j$ for $\forall\, i, j$, it is equivalent to

$$\begin{bmatrix} \mathbf{0}^\top & -w'_i\mathbf{x}_i^\top & y'_i\mathbf{x}_i^\top \\ w'_i\mathbf{x}_i^\top & \mathbf{0}^\top & -x'_i\mathbf{x}_i^\top \\ -y'_i\mathbf{x}_i^\top & x'_i\mathbf{x}_i^\top & \mathbf{0}^\top \end{bmatrix} \begin{pmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{pmatrix} = \mathbf{0}. \tag{A.1.3}$$

Note that only two equations are linearly independent (sum of x'_i times the first equation, y'_i times the second and w'_i times the third is zero) among the three of the equations in (A.1.3). Thus (A.1.3) reduces to

$$\begin{bmatrix} \mathbf{0}^\top & -w'_i\mathbf{x}_i^\top & y'_i\mathbf{x}_i^\top \\ w'_i\mathbf{x}_i^\top & \mathbf{0}^\top & -x'_i\mathbf{x}_i^\top \end{bmatrix} \begin{pmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{pmatrix} = \mathbf{0}. \tag{A.1.4}$$

This will be written

$$A_i\mathbf{h} = \mathbf{0}$$

where A_i is the 2×9 matrix in (A.1.4) and

$$\mathbf{h} = \begin{pmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{pmatrix}, \quad H = \begin{pmatrix} \mathbf{h}^{1\top} \\ \mathbf{h}^{2\top} \\ \mathbf{h}^{3\top} \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix}.$$

Each point correspondence gives a set of two linearly independent equations $A_i\mathbf{h} = 0$ in the entries of H. Given four point correspondences, we have a matrix equation

$$A\mathbf{h} = \mathbf{0} \tag{A.1.5}$$

where A is the 8×9 matrix obtained by concatenating A_i's vertically. Although the above matrix equation has a trivial solution $\mathbf{h} = \mathbf{0}$, it is of no interest to us since a homography matrix is non-zero. In the practical case, the coefficient matrix A is of rank 8 so that its null space is 1-dimensional. Thus, there exist non-trivial solutions for (A.1.5) which are non-zero vectors in the null space of A. Since a point in projective space is determined up to non-zero scalar multiplication, a homography between two projective spaces is also determined up to non-zero scalar multiplication. Therefore, we may choose any non-zero vector in the null space of A for our solution. For simplicity, we find the vector

$$\mathbf{h} = (h_1, \ldots, h_8, 1)^\top$$

whose 9-th entry is 1 in the null space of A and take the matrix

$$H = \begin{pmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & 1 \end{pmatrix}. \tag{A.1.6}$$

for a desired homography matrix H.

A.2 Optimization of the Cost Function

In this section, we describe the optimization of the cost function (4.3.2) for the 3D reconstruction algorithm using the least-squares method [10].

Objective Minimize the below cost function L

$$L(\{z_{ij}\}_{i,j=0}^{m,n}) = \sum_{i,j=0}^{m-1,n-1} \|z_{i+1,j}\mathbf{p}_{i+1,j} - z_{i+1,j+1}\mathbf{p}_{i+1,j+1} + z_{i,j+1}\mathbf{p}_{i,j+1} - z_{ij}\mathbf{p}_{ij}\|^2. \tag{A.2.1}$$

under the constraint

$$\|Z\| = \|(z_{00}, \ldots, z_{mn})^\top\| = 1$$

where $\mathbf{p}_{ij} = (x_{ij}, y_{ij}, 1)^\top$ is an (i, j)-th corner point of the grid G on the document image plane.

APPENDIX A.

For each $i = 0, \ldots, m-1$ and $j = 0, \ldots, n-1$, let A_{ij} be a $3 \times (m+1)(n+1)$ matrix

$$A_{ij} = \begin{bmatrix} \cdots & \mathbf{p}_{ij} & -\mathbf{p}_{i,j+1} & \cdots & -\mathbf{p}_{i+1,j} & \mathbf{p}_{i+1,j+1} & \cdots \end{bmatrix}$$

whose k-th column $[A_{ij}]^k$ is

$$[A_{ij}]^k = \begin{cases} \mathbf{p}_{ij}, & \text{if } k = i*(n+1)+j+1, \\ -\mathbf{p}_{i,j+1}, & \text{if } k = i*(n+1)+j+2, \\ -\mathbf{p}_{i+1,j}, & \text{if } k = (i+1)*(n+1)+j+1, \\ \mathbf{p}_{i+1,j+1}, & \text{if } k = (i+1)*(n+1)+j+2, \\ \mathbf{0}, & \text{else.} \end{cases}$$

If we write

$$A = \begin{bmatrix} A_{00} \\ A_{01} \\ \vdots \\ A_{m-1,n-1} \end{bmatrix}, \quad Z = \begin{pmatrix} z_{00} \\ z_{01} \\ \vdots \\ z_{mn} \end{pmatrix} \tag{A.2.2}$$

then the cost function (A.2.1) can be written as

$$L(Z) = \|AZ\|^2 \tag{A.2.3}$$

where the norm above is the Frobenius norm. So the minimization problem reduces to find Z minimizing $\|AZ\|^2$. The cost function of this form can be minimized using the singular value decomposition (SVD) up to global scaling. Since we fixed the scaling by constraint $\|Z\| = 1$, we can find the unique optimal solution as follows. To begin with, we state the SVD of a matrix.

Singular Value Decomposition (SVD) Any $m \times n$ real matrix A can be factored into

$$A = UDV^\top \tag{A.2.4}$$

where U is an $m \times m$ orthogonal matrix, D is an $m \times n$ rectangular diagonal matrix with non-negative real diagonal entries, and V is an $n \times n$ orthogonal matrix.

APPENDIX A.

Using a SVD of a matrix, we can minimize (A.2.3). Let A be the matrix in (A.2.2) and $A = UDV^\top$ be a SVD of A. A is a $3mn \times (m+1)(n+1)$ matrix and A has generally more rows than columns in the proposed method. Due to the noise, A is of full rank $(m+1)(n+1)$ so that the $3mn \times (m+1)(n+1)$ rectangular diagonal matrix D is of the form

$$D = \begin{bmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_{(m+1)(n+1)} \\ \hline & & \mathbf{0} & \end{bmatrix}$$

with $d_1 \geq d_2 \geq \cdots \geq d_{(m+1)(n+1)} > 0$. Then,

$$L(Z) = \|AZ\|^2 = \|UDV^\top Z\|^2 = \|DV^\top Z\|^2$$

since U is an orthogonal matrix. Moreover, if we write $W = V^\top Z$,

$$L(Z) = \|DV^\top Z\|^2 = \|DW\|^2$$

is minimized when $W = (0, \ldots, 0, 1)^\top$. Thus, the optimal solution is

$$Z = (VV^\top)Z = VW = [V]^{(m+1)(n+1)},$$

the last column of the orthogonal matrix V. Consequently, from the minimizing solution $Z = (z_{00}, \ldots, z_{mn})^\top$, we find the scene points $\tilde{\mathbf{p}}_{ij} = z_{ij}\mathbf{p}_{ij}$'s of the image points $\mathbf{p}_{ij}$.

Bibliography

[1] Bradley, D., Roth, G., *Adapting Thresholding Using the Integral Image*, Journal of Graphics Tools. Vol. 12, No. 2, (2007), pp.13–21.

[2] Brown, M. S. and Tsoi, Y. -., *Distortion Removal For Camera-Imaged Print Materials Using Boundary Interpolation*, International Journal of Image and Graphics, 5(02), (2005), pp. 311-328.

[3] Brown, M. S. and Tsoi, Y. -., *Geometric and shading correction for images of printed materials using boundary*, in IEEE Transactions on Image Processing, vol. 15, no. 6, (2006), pp. 1544-1554, doi: 10.1109/TIP.2006.871082.

[4] Brown, M. S. and Tsoi, Y. -., *Multi-View Document Rectification using Boundary*, 2007 IEEE Conference on Computer Vision and Pattern Recognition, (2007), pp. 1-8, doi: 10.1109/CVPR.2007.383251.

[5] Cao, H., Ding, X., and Liu, C., *A cylindrical surface model to rectify the bound document image*, Proceedings Ninth IEEE International Conference on Computer Vision, (2003), pp. 228-233 vol.1, doi: 10.1109/ICCV.2003.1238346.

[6] Chaki, N., Shaikh, S.H., Saeed, K. *A Comprehensive Survey on Image Binarization Techniques*. Exploring Image Binarization Techniques. Studies in Computational Intelligence, vol 560. Springer, (2014), pp. 5-15.

BIBLIOGRAPHY

[7] Dasgupta, T., Das, N., Nasipuri, M., *Multistage Curvilinear Coordinate Transform Based Document Image Dewarping using a Novel Quality Estimator*, arXiv, 2020, arXiv:2003.06872.

[8] Fu, B., Li, W., Wu, M., Li, R., Xu, Z., *A document rectification approach dealing with both perspective distortion and warping based on text flow curve fitting*, Int J Image Graphics 12(01):1250002., 2012, https://doi.org/10.1142/S0219467812500027

[9] Hartley, R. *In defense of the eight-point algorithm*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 6, (1997), pp. 580-593.

[10] Hartley, R., Zisserman, A. *Multiple View Geometry in Computer Vision*, Cambridge, U.K.: Cambridge University Press, 2003.

[11] He, Y., Pan, P., Xie, S., Sun, J. and Naoi, S., *A Book Dewarping System by Boundary-Based 3D Surface Reconstruction*, 2013 12th International Conference on Document Analysis and Recognition, (2013), pp. 403-407, doi: 10.1109/ICDAR.2013.88.

[12] Jiang, X., Long, R., Xue, N., Yang, Z., Yao, C., and Xia, G.-S., *Revisiting Document Image Dewarping by Grid Regularization*, 2022.

[13] Kil, T., Seo, W., Koo, H.I., and Cho, N.I. *Robust Document Image Dewarping Method Using Text-Lines and Line Segments*, 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), (2017), pp. 865-870.

[14] Kim, B.S., Koo, H.I., and Cho, N.I. *Document dewarping via textline based optimization*, Pattern Recognition, vol. 48, no. 11, (2015), pp. 3600–3614.

[15] Koo, H.I. *Text-line detection in camera-captured document images using the state estimation of connected components.* IEEE Trans. Image Process. 25(11), (2016), pp. 5358–5368.

[16] Koo, H.I., Cho, N.I. *State Estimation in a Document Image and Its Application in Text Block Identification and Text Line Extraction*, Computer Vision – ECCV 2010. ECCV 2010. Lecture Notes in Computer Science, vol 6312. Springer, Berlin, Heidelberg., (2010), pp. 421-434.

[17] Liang, J., DeMenthon, D., and Doermann, D., *Flattening curved documents in images*, 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), (2005), pp. 338-345 vol. 2, doi: 10.1109/CVPR.2005.163

[18] Lokhande, S.S., Dawande, N.A. *A Survey on Document Image Binarization Techniques* 2015 International Conference on Computing Communication Control and Automation, (2015), pp. 742-746.

[19] Otsu, N., *A Threshold Selection Method from Gray-Level Histograms*, IEEE Transactions on Systems, Man, and Cybernetics. Vol. 9, No. 1, (1979), pp. 62–66.

[20] Saha, S., Basu, S., Nasipuri, M. *Automatic Localization and Recognition of License Plate Characters for Indian Vehicles*, International Journal of Computer Science and Emerging Technologies, (IJCSET), vol. 2, no. 4, (2011), pp. 520-533.

[21] Sauvola, J., Pietikäinen, M. *Adaptive document image binarization*, Pattern Recognition, Volume 33, Issue 2, (2000), pp. 225-236.

[22] Stamatopoulos, N., Gatos, B., Pratikakis I. and Perantonism, S. J., *A Two-Step Dewarping of Camera Document Images*, 2008 The Eighth IAPR International Workshop on Document Analysis Systems, (2008), pp. 209-216, doi: 10.1109/DAS.2008.40

[23] Stathis, P., Kavallieratou, E., Papamarkos, N. *An evaluation survey of binarization algorithms on historical documents*, 2008 19th International Conference on Pattern Recognition, (2008), pp. 1-4.

[24] Tian, Y., Narasimhan, S.G. *Rectification and 3D Reconstruction of Curved Document Images*, Proceedings of the IEEE Computer Society

Conference on Computer Vision and Pattern Recognition, (2011), pp. 377-384.

[25] Wagdy, M., Amin, K., and Ibrahim, M. *Dewarping Document Image Techniques: Survey and Comparative Study*, International Journal of Image and Graphics Vol. 21, No. 3, 2021, 2150031.

[26] Wu, M., Li, R., Li, W., Heaney Jr, E. P., Chan, K. M., Rapelje, K. A., Fu, B., and Feng, B. *Model-based dewarping method and apparatus* U.S. Patent No. 9,805,281. Washington, DC: U.S. Patent and Trademark Office., 2017

국문초록

최근에는 대부분의 스캔된 이미지들이 전통적인 평판스캐너가 아닌 카메라, 스마트폰, 태블릿 PC 등의 휴대기기들로부터 얻어진다. 이전 스캐너들의 스캐닝 과정과는 다르게 휴대기기들을 이용한 이미지 캡쳐링 과정은 원근왜곡, 종이의 접힘으로 인한 왜곡, 그리고 종이의 휘어짐으로 인한 왜곡 등 다양한 왜곡들을 수반할 수 있다. 이 논문에서는 이러한 왜곡들을 제거할 수 있는 문서 경계와 3차원 재구성에 기반한 강력한 디워핑 방법을 제안하고자 한다. 첫번째 방법에서는, 문서 경계를 이용하여 문서 이미지 위에 곡선으로 이루어진 그리드를 만들고, 3차원 공간 상의 문서 곡면을 재구성한다. 그리고 재구성된 문서 곡면으로부터 계산된 국소적 호모그래피들을 이용하여 이미지를 수정한다. 우리가 제안하는 방법의 몇몇 단계는 다른 연구에서 개별적으로 사용된 경우도 있지만, 우리는 전체적인 과정에서 안정성을 높이는 동시에 각 방법의 장점들을 이용하고 조합하여 강력한 디워핑 방법을 제안한다. 이에 더하여, 우리는 왜곡된 텍스트 영역의 경계를 수정하여 전체적인 과정을 보완하였고, 이 절차를 간결하고, 직관적이며, 강력하면서도 좋은 결과를 내는 독립적인 디워핑 방법으로 개발하였다.

주요어휘: 문서 이미지, 평판화, 문서 경계, 그리드, 3차원 재구성
학번: 2018-29556

감사의 글

우선 학위 과정동안 저에게 아낌없는 가르침을 주시고 연구자로서 모범이 되는 모습을 보여주신 현동훈 지도교수님께 깊은 감사의 말씀을 전합니다. 석사 학위 과정 동안 교수님의 지도 아래서 여러 분야에 걸친 연구를 접해보고 참여해보면서 다양한 경험을 하고 학위 과정을 마칠 수 있었습니다. 앞으로 있을 박사 학위 과정에서도 교수님께 받은 지도를 기억하며 좋은 연구를 위해 노력하도록 하겠습니다. 또한 논문심사를 맡아주시고 심사 과정에서 아낌없는 조언을 주신 천정희 교수님, 국웅 교수님께도 감사의 말씀을 드립니다.

마지막으로 언제나 저의 버팀목이 되어주신 부모님께 감사하고 사랑한다는 말씀과 함께 본 학위과정을 무사히 마친 공을 돌립니다. 이후 박사 학위 과정도 무사히 마치고 부모님께 좋은 모습 보여드릴 수 있도록 노력하겠습니다.