공학박사 학위논문

Document Image Dewarping and Scene Text Rectification based on Alignment Properties

정렬 특성들 기반의 문서 및 장면 텍스트 영상 평활화 기법

2017 년 8 월

서울대학교 대학원

전기컴퓨터공학부

길 태 호

ABSTRACT

The optical character recognition (OCR) of text images captured by cameras plays an important role for scene understanding. However, the OCR of camera-captured image is still considered a challenging problem, even after the text detection (localization). It is mainly due to the geometric distortions caused by page curve and perspective view, therefore their rectification has been an essential pre-processing step for their recognition. Thus, there have been many text image rectification methods which recover the fronto-parallel view image from a single distorted image. Recently, many researchers have focused on the properties of the well-rectified text. In this respect, this dissertation presents novel alignment properties for text image rectification, which are encoded into the proposed cost functions. By minimizing the cost functions, the transformation parameters for rectification are obtained. In detail, they are applied to three topics: document image dewarping, scene text rectification, and curved surface dewarping in real scene.

First, a document image dewarping method is proposed based on the alignments of text-lines and line segments. Conventional text-line based document dewarping methods have problems when handling complex layout and/or very few text-lines. When there are few aligned text-lines in the image, this usually means that photos, graphics and/or tables take large portion of the input instead. Hence, for the robust document dewarping, the proposed method uses line segments in the image in addition to the aligned text-lines. Based on the assumption and observation that all the transformed line segments are still straight (line to line mapping), and many of them are horizontally or vertically aligned in the well-rectified images, the proposed method encodes this properties into the cost function in addition to the text-line based cost. By minimizing the function, the proposed method can obtain transformation parameters for page curve, camera pose, and focal length, which are used for document image rectification. Considering that there are many outliers in line segment directions and miss-detected text-lines in some cases, the overall algorithm is designed in an iterative manner. At each step, the proposed method removes the text-lines and line segments that are not well aligned, and then minimizes the cost function with the updated information. Experimental results show that the proposed method is robust to the variety of page layouts.

This dissertation also presents a method for scene text rectification. Conventional methods for scene text rectification mainly exploited the glyph property, which means that the characters in many language have horizontal/vertical strokes and also some symmetric shapes. However, since they consider the only shape properties of individual character, without considering the alignments of characters, they work well for only images with a single character, and still yield mis-aligned results for images with multiple characters. In order to alleviate this problem, the proposed method explicitly imposes alignment constraints on rectified results. To be precise, character alignments as well as glyph properties are encoded in the proposed cost function, and the transformation parameters are obtained by minimizing the function. Also, in order to encode the alignments of characters into the cost function, the proposed method separates the text into individual characters using a projection profile method before optimizing the cost function. Then, top and bottom lines are estimated using a least squares line fitting with RANSAC. Overall algorithm is designed to perform character segmentation, line fitting, and rectification iteratively. Since the cost function is non-convex and many variables are involved in the function, the proposed method also develops an optimization method using Augmented Lagrange Multiplier method. This dissertation evaluates the proposed method on real and synthetic text images and experimental results show that the proposed method achieves higher OCR accuracy than the conventional approach and also yields visually pleasing results.

Finally, the proposed method can be extended to the curved surface dewarping in real scene. In real scene, there are many circular objects such as medicine bottles or cans of drinking water, and their curved surfaces can be modeled as Generalized Cylindrical Surfaces (GCS). These curved surfaces include many significant text and figures, however their text has irregular structure compared to documents. Therefore, the conventional dewarping methods based on the properties of well-rectified text have problems in their rectification. Based on the observation that many curved surfaces include well-aligned line segments (boundary lines of objects or barcode), the proposed method rectifies the curved surfaces by exploiting the proposed line segment terms. Experimental results on a range of images with curved surfaces of circular objects show that the proposed method performs rectification robustly.

Key words: document image dewarping, scene text rectification, character segmentation, curved surface dewarping

Student number: 2013-30220

Contents

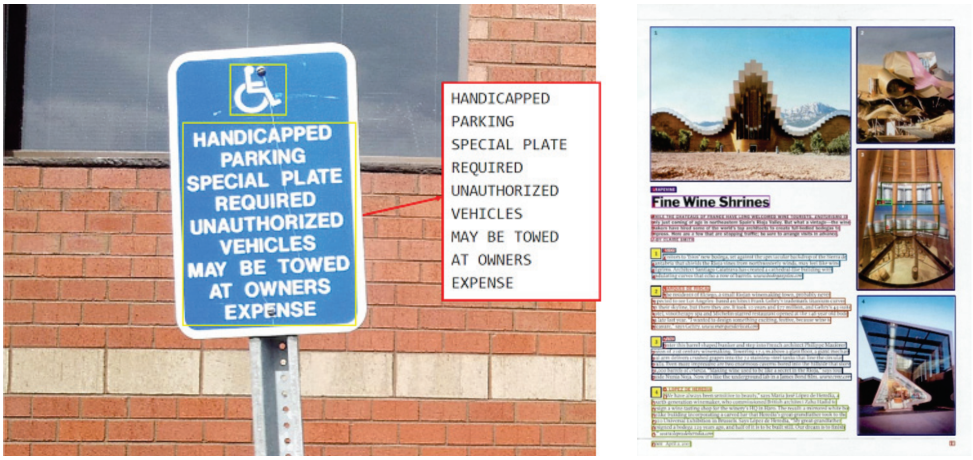
List of Figures

List of Tables

Chapter 1

Introduction

The analysis of text in the scene such as optical character recognition (OCR) and document segmentation helps their scene understanding. For instance, the text on the road sign provides drivers with traffic information, and the document contains contents of the book, as shown in Fig. 1.1. The various methods have been studied for the text analysis [7–9], these methods can be divided into two topics: document and scene text analysis. In the document analysis, numerous methods have been proposed for the scanned document image processing (printed documents are converted to digital images with flatbed scanners and document image processing algorithms are applied) [7,8,10,11]. However, with the recent development of smart-phones having high-resolution digital cameras, document image processing algorithms are required to handle camera-captured images as well as scanned documents [12,13]. Also, in the scene text, numerous methods have been proposed for the scene text analysis such as detection and recognition in the camera-captured images [7,9]. In summary, It is important issue to perform text analysis in camera-captured images for both document and general scene. However, the text analysis of camera-captured images is



(a) (b)

Figure 1.1: Examples of the text analysis. (a) Scene text detection and recognition, (b) Document image segmentation.

considered a challenging task due to the geometric distortions caused by page curve and camera view, and hence their rectification which recovers flat fronto-parallel view images from distorted images is an essential pre-processing step for the text analysis.

In this dissertation, novel alignment properties for text image rectification are presented, and they are applied to three topics: document image dewarping, scene text rectification, and curved surface dewarping in real scene. Unlike previous text-line based methods for document dewarping, the proposed approach exploits line segments in the image in addition to the aligned text-lines. A document image dewarping method of the proposed approach can obtain the rectification transformation that removes the distortions in both text and non-text regions. The proposed approach is expanded to scene text rectification research by exploiting the alignments of characters as well as the glyph property (this dissertation uses conventional low-rank transform [4] for the glyph property), it effectively rectifies the distorted image having severe perspective distortions with considering the alignments of characters. In addition, the proposed method is extended to the curved surface dewarping in real scene. The proposed method transforms curved surface images to flat images using the properties of line segments.

The main contributions of this dissertation are summarized as:

- A robust document image dewarping algorithm for various layouts is proposed by using the alignments of text-lines and line segments.
- A scene text rectification algorithm is proposed using two different properties that are based on the low-rank assumption and character alignments.
- The proposed alignment term can be extended to the curved surface dewarping in real scene.

1.1 Document image dewarping

In this dissertation, a document image dewarping method is proposed based on the alignments of text-lines and line segments. For the single document image dewarping (without additional information), numerous methods using text-lines have been proposed. Although these text-line based methods are able to reduce geometric distortions without the additional information, they focus on text regions and sometimes yield severe distortions on non-text regions (e.g., photos, graphics or tables). In summary, text-line based methods exploited regular structures of text-lines and text-blocks, and they basically work for text-abundant cases. In order to alleviate the limitations of text-line based methods, a dewarping method that exploits the properties of text and non-text regions is present: The proposed method uses line

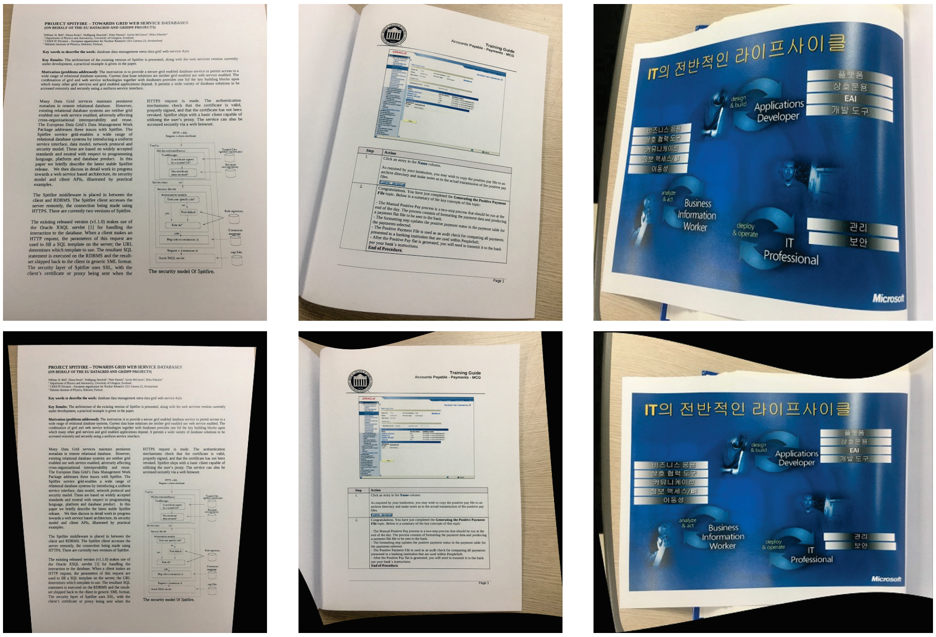

PROJECT SPITFIRE – TOWARDS GRID WEB SERVICE DATABASES
Training Guide
Accounts Payable - Payments - MCG
IT의 전반적인 라이프사이클
design & build
Applications Developer
analyze & act
Business Information Worker
deploy & operate
IT Professional
Microsoft

Figure 1.2: Examples of the document image dewarping. the first row: input camera-captured document images, the second row: document image dewarping results of inputs.

segments as well as text-lines.

First, the lines extracted in the curved document surface are still straight lines in the well-rectified image (line to line mapping). Also, since non-text regions in documents usually have many line segments that are horizontally or vertically aligned in the well-rectified images (e.g., tables and the boundaries of images), the proposed method encodes this two properties into the proposed cost function, as well as the properties of text-lines. Considering that there are many outliers in line segment directions and miss-detected text-lines (false positive) in some cases, the overall algorithm is designed in an iterative manner. At each step, the proposed method refines text-lines and line segments by removing the text-lines and line segments that are not well aligned, and then minimizes the cost function with the updated informa-

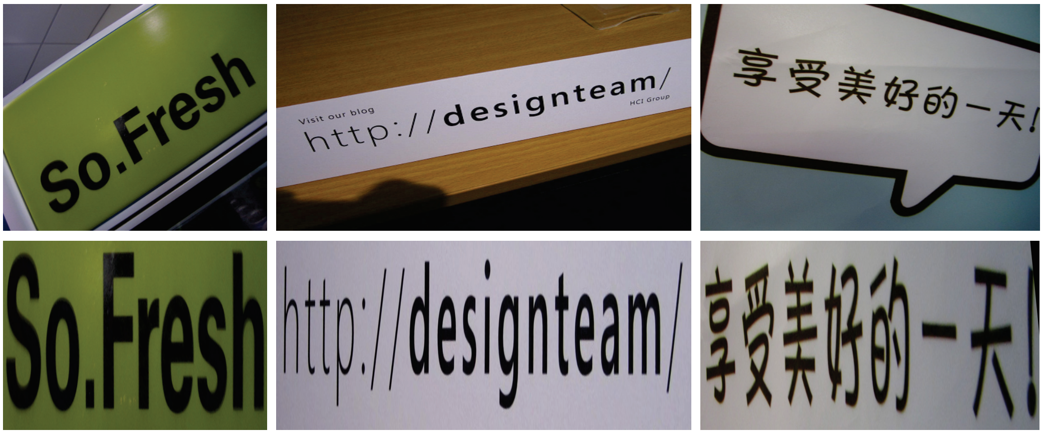


Figure 1.3: Example of the scene text image rectification. the first row: input camera-captured scene text images, the second row: scene text rectification results of inputs.

tion. The cost function is minimized via the Levenberg-Marquardt algorithm [14,15] and the proposed method can obtain the rectification transformation that removes the distortions in both text and non-text regions.

1.2 Scene text rectification

In general, scene text rectification faces some challenges. First, the scene text image contains a few characters compared to the document image. Many valuable information can be extracted in the case of text-abundant document images, whereas there are few features to be extracted in the case of scene text images. Second, there are too many variants in the character shapes, (mixed) languages and stroke widths. Hence, it is difficult to rectify the scene without these prior information. Finally, dealing with the perspective distortion is not straightforward. Generally, when the text is on a planar surface, its transformation to the fronto-parallel view can be modeled as a projective transformation. In this transformation, there are four parameters related with skew, shearing, horizontal and vertical foreshortening, where estimating these parameters simultaneously is a challenging problem.

Considering the problems caused by these difficulties, this dissertation proposes a new scene text rectification algorithm that exploits character alignment constraints in addition to other properties employed in the conventional works. Specifically, the existing methods for scene text rectification mainly exploited the glyph property which is a common shape property for the undistorted characters. Some examples of glyph property are that the characters in many languages have horizontal and/or vertical strokes and many characters have some symmetries in their shapes. When the character or set of characters is represented as a matrix, the rank of the matrix for the well-rectified character is usually lower than that for the distorted ones.

However, glyph property based methods still yield mis-aligned results for multiple characters, since they do not consider the alignments of characters. Thus, the proposed method uses alignments of characters in addition to the conventional glyph properties. The proposed method designs a cost function including these two properties, the minimization of which provides the transformation parameters. In order to encode the alignment property into the cost function, the proposed method needs to segment the text into individual characters. The character segmentation is relatively easy when the text is in fronto-parallel view without any perspective distortion, but it is also a difficult problem in the case of distorted images [16, 17]. In short, the rectification and character segmentation are a chicken-and-egg problem, i.e., better segmentation needs better rectification and vice versa. The proposed method solves this problem by performing the character segmentation and rectification iteratively. Since the cost function is non-convex and many variables including alignments are involved in the function, it is not straightforward to minimize the cost function.

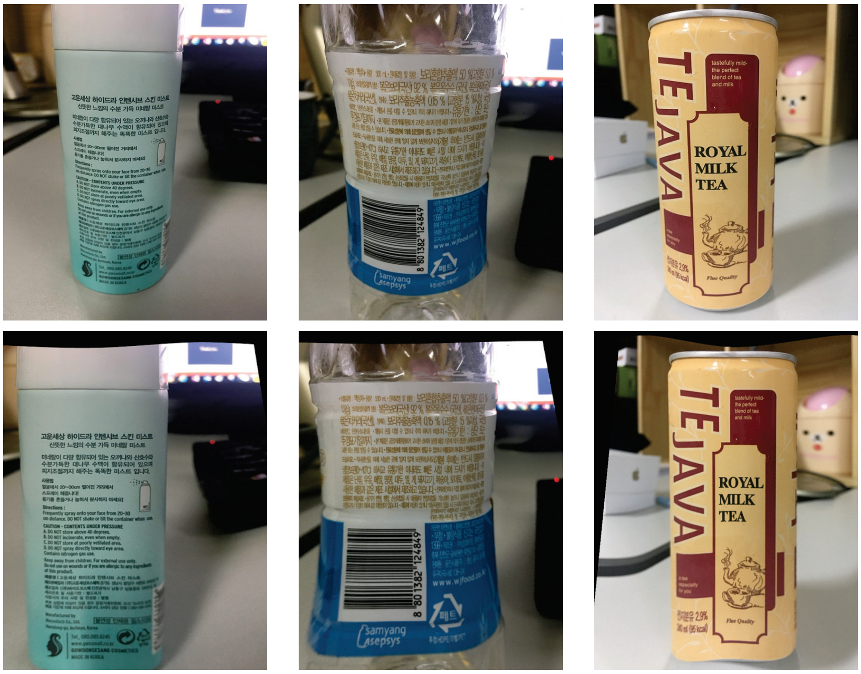


Figure 1.4: Examples of the curved surface dewarping in real scene. the first row: input camera-captured curved surfaces in real scene, the second row: dewarping results of inputs.

To solve this problem, the proposed method adds the auxiliary variables and finds solution by solving the linearized problem iteratively. Then, the proposed method can obtain the rectification transformation that removes the perspective distortions of scene text.

1.3 Curved surface dewarping in real scene

In this dissertation, the proposed document dewarping method can be extended to the dewarping of circular surface in real scene. In real scene, there are many circular objects such as medicine bottles or cans of drinking water. These circular objects includes many significant text and figures, however their text has irregular structure compared to documents. Therefore, the conventional dewarping methods based on the properties of well-rectified text have problems in their rectification. Since many circular objects include well-aligned line segments (boundary lines of objects or barcode), the proposed method rectifies the circular surface by exploiting proposed line segment terms. The proposed method can obtain the rectification transformation that removes the geometric distortions of curved surface image.

1.4 Contents

In chapter 2, this dissertation presents related works that are the reviews of the conventional approaches for document image dewarping, scene text rectification and curved surface dewarping in real scene. The proposed method for the document image dewarping is introduced in chapter 3. A document image dewarping method using line segment alignment properties is explained in details. Then, the proposed approach for the scene text rectification is introduced in chapter 4. A scene text rectification method using the glyph and character alignment properties is explained in details, and following experimental results are also presented. Then, the application is presented in 5. By exploiting the alignments of line segments, curved surface images captured in real scene are rectified. Finally, this dissertation is concluded in chapter 6.

Chapter 2

Related work

Numerous methods have been proposed for text image rectification. In this chapter, this dissertation reviews three related topics: document image dewarping, scene text rectification, and curved surface dewarping in real scene.

2.1 Document image dewarping

2.1.1 Dewarping methods using additional information

For the document image dewarping, many methods were developed by using depth measuring hardwares (e.g., structured light or laser scanners) [18–21]. This approach is able to estimate the surfaces of curved pages very effectively, however, the requirements of special hardwares limit their application areas. In [22–24], curved pages were estimated from multiple images taken from different viewpoints. Although they could perform rectification without additional hardwares, taking multiple images are burdensome for common users and their computation complexity is also very high. In [25–27], the shape-from-shading approach exploiting illumination conditions was proposed. Although these methods can be applied to a single document image, their assumptions on illumination may not hold in many situations.

In summary, the above mentioned methods have limitations that need additional informations such as page surface model acquired by the special hardwares, multiple images or the assumption on illumination.

2.1.2 Text-line based dewarping methods

For the single document image dewarping (without additional information), numerous methods using text-lines have been proposed. Since text-lines are common and show regular structures in document images, they are considered very useful features in the document image dewarping.

In [28], two vanishing points were estimated by many horizontal (made by text-lines) and vertical lines (made by line feeds). This approach removes effectively perspective distortions, however is not suitable for geometric distortions by curved surfaces. In most of text-line based methods, curved surfaces are modeled with the generalized cylindrical surface (GCS) [22] and the shapes are estimated from the properties of text-lines. In [29–31], curved page surfaces were estimated by fitting top and bottom text-lines to flat document regions. In [1], the properties of text-lines (in undistorted documents) were encoded into a cost function, and curved page surfaces and camera pose were estimated by minimizing the function. Although these text-line based methods are able to reduce geometric distortions without the additional information, they focus on text regions and sometimes yield severe distortions on non-text regions.

In summary, text-line based methods exploited regular structures of text-lines and text-blocks, and they basically work for text-abundant cases.

2.2 Scene text rectification

For the rectification of camera-captured document images, many methods have been proposed based on the Hough transform [32, 33], distance transform [34], gradient directions [35], vanishing point estimation [36], texture flow fields [37], text alignment properties [1], and multiple view approaches [23]. However, they exploited abundant text, text-lines, or text blocks, which is not the case for the scene text images.

For the scene text rectification, projection profile based methods have been studied [38, 39]. The skew angle was estimated using projection profiles [38]. In [39], top and baselines, and shearing angles were estimated by projection-based method. They removes effectively the skew and shearing distortions, however are not suitable for general perspective distortions.

Recently methods exploiting the glyph property of individual characters were also proposed. A skew estimator that exploits intuitive glyph properties was proposed in [40]. However, this method focused on skew and shearing distortions which are not suitable for general perspective distortions. In [4], the homography for rectification was estimated using a low-rank transform. This method exploited the low-rank property indicating that a variety of objects have symmetry textures (such as building facade and repeated pattern). According to the experiments, it was shown to be very effective for rectifying a single character. However, it is difficult to impose the character alignment constraints only by using the low rank approach. The methods that further improved the low-rank approach were proposed in [41, 42], which especially improved the case of multiple characters. However, these methods do not work well without prior information about the separation of characters. In [43], Wang et al. assumed that the line segments on the characters pass through the vanishing points, and developed a method that removes perspective distortions by using the estimated vanishing points [43]. However, this assumption does not hold for many characters (e.g., ‘x’, ‘y’, and ‘o’).

For the alignments of characters, skew angles were estimated by fitting the straight lines to the top and bottom points of the text and shearing angles were also estimated by performing a linear regression on the shear variation [44]. This method separated text into individual characters in distorted input images, before rectification. Since the character segmentation is difficult problem in the case of distorted images, this problem reduces rectification performance.

In [45], perspective and curved distortions were corrected by a Spatial Transformer Network (STN), and rectified text was recognized by a Sequence Recognition Network (SRN). However, it is not clear whether this approach can be applied to scene text rectification. It seems that the rectified results still have some distortions, and they do not use standard OCR system.

2.3 Curved surface dewarping in real scene

In computer vision, recovering 2D flat image from curved surface, while estimating the 3D object shape, has been an important issue. In the past decades, in order to recover 2D flat surface, the structure from motion approach (SfM) has been studied [46, 47]. This approach estimates 3D structures from 2D image sequences that are correlated by motions. This approach effectively estimates the 3D shape of objects, however it needs to multiple images. For recovering flat surface of a single image, additional assumptions for the object surface are needed. For this, many methods have been proposed based on the regular/symmetrical lattice structure [48, 49], and low-rank transform [6]. This approach effectively estimates the 3D shape using the assumption, however they can be adopted to only repeated pattern like lattice.

Chapter 3

Document image dewarping

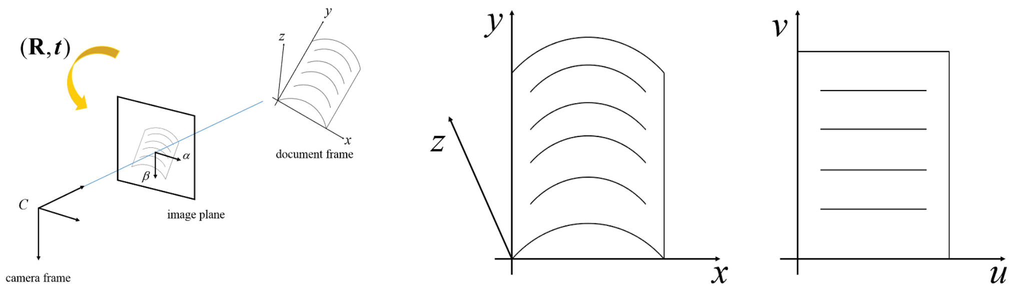
3.1 Proposed cost function

The proposed cost function for document image dewarping is presented in this section. First, a parametric model for the rectification transformation is introduced, and the proposed cost function reflecting the properties of line segments is presented.

3.1.1 Parametric model of dewarping process

For the parametric modeling of the dewarping process, the proposed method adopts the model in [1]. Given a document surface as shown in Fig. 3.1-(a), a point (α, β) on an image domain corresponds to a point (x, y, z) on the curved document surface with the relation

$$k \begin{pmatrix} \alpha \\ \beta \\ 1 \end{pmatrix} = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix} \left(s\mathbf{R}^\top \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \boldsymbol{t} \right), \tag{3.1}$$



(a) (b) (c)

Figure 3.1: Dewarping model. (a) A curved document viewed by a camera, (b) Curved document coordinate, (c) Flat document coordinate.

where k is a scale for homogeneous coordinate, f is the focal length of a camera, (c_x, c_y) is an image center and $(s, \mathbf{R}, \boldsymbol{t})$ are scale, rotation and translation between two frames, respectively. Since the parameters s and $\boldsymbol{t}$ are related to not rectification but image scale and resolution, the proposed method sets $s = 1$ and $\boldsymbol{t} = [0, 0, f]^\top$ without loss of generality.

For a GCS model as shown in Fig. 3.1-(b) and (c), a point on a curved surface can be transformed to a corresponding point (u, v) on the rectified document with

$$\begin{aligned} u &= \int_0^x \sqrt{1 + g'(t)^2} dt, \\ v &= y, \end{aligned} \tag{3.2}$$

where $g(x)$ is the document surface equation that is represented with a polynomial.

$$z = g(x) = \sum_{m=0}^{M} a_m x^m, \tag{3.3}$$

When the number of polynomial parameters M is low, polynomial equation cannot represent the page curl. On the other hand, when the parameter M is very high, fitting parameters are over-fitted. After extensive experiments, the proposed method

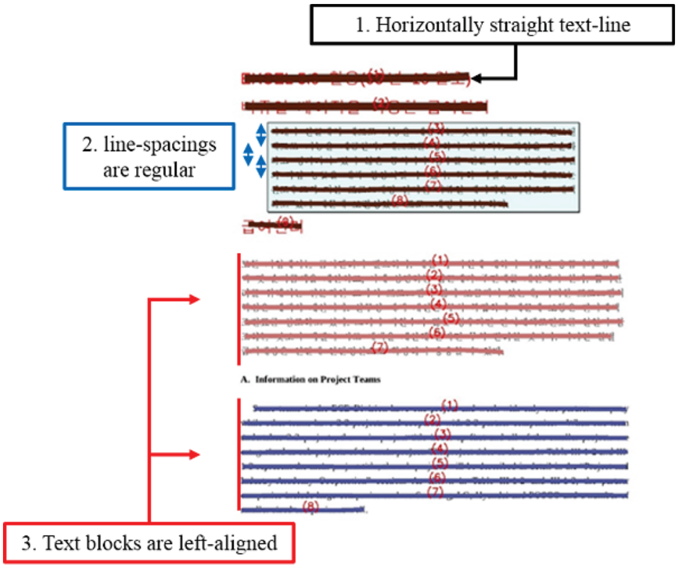

1. Horizontally straight text-line
2. line-spacings are regular
A. Information on Project Teams
3. Text blocks are left-aligned

Figure 3.2: Illustration of text-line properties in well-rectified document image: 1. Text-lines are horizontally straight lines, 2. (Blue) line-spacings between two neighboring text-lines are regular, 3. Text-blocks are left-aligned (in red lines).

confirms that fourth (M=4) polynomial equation is sufficient to represent page curl (like [1, 50]).

By combining (3.1), (3.3), and (3.3), points on the image domain can be transformed to the corresponding points on rectified document images.

In summary, the geometric relation between the captured image domain and the rectified document domain can be parameterized with polynomial parameters $\{a_m\}_{m=0}^{M}$, camera pose $\mathbf{R}$ and camera focal length f. Therefore, the document image rectification process can be formulated as an estimation problem of polynomial parameters, camera pose and focal length.

3.1.2 Cost function design

For the estimation of the dewarping parameters $\Theta = (\{a_m\}_{m=0}^{M}, \mathbf{R}, f)$, the proposed method develops a cost function:

$$f_{cost}(\Theta) = f_{text}(\Theta) + f_{line}(\Theta) + \lambda_1 f_{regular}(f), \tag{3.4}$$

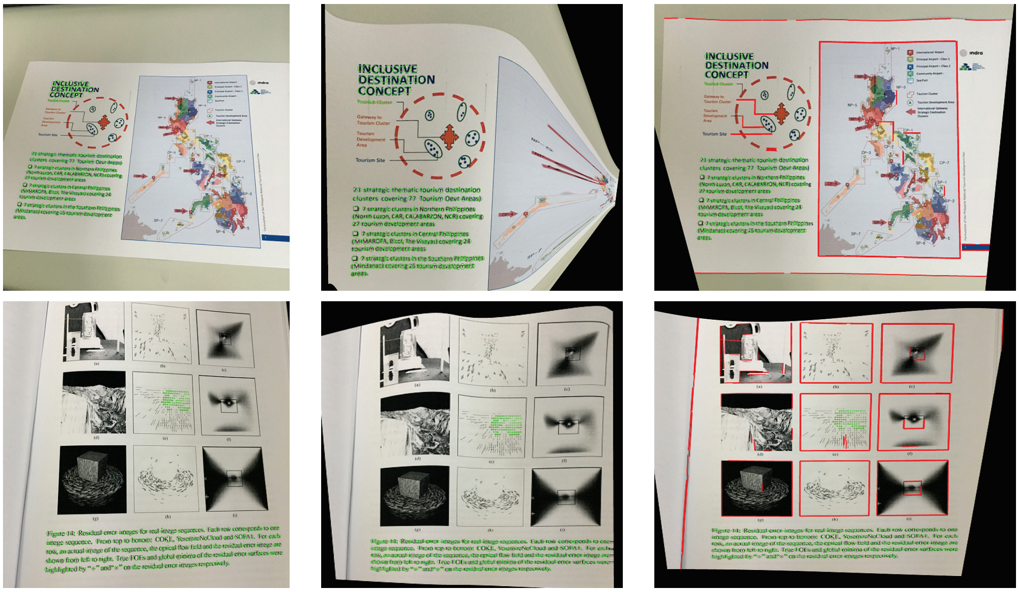
where $f_{text}(\Theta)$ is a term reflecting the properties of text-lines in rectified images [1]. To be precise, they first extract text-lines in the distorted document images [50], then design the cost function $f_{text}(\Theta)$ that becomes small when transformed text-lines are well-aligned: horizontally straight, line-spacings between two neighboring text-lines are regular, and text-blocks are either left-aligned, right-aligned, or justified, as shown in 3.2.

However, this term focuses on text regions and sometimes yields severe distortions on non-text regions (e.g., photos, graphics or tables) as shown in Fig. 3.3. In order to alleviate the limitations of text-line based methods, the proposed method exploits the properties of both text and non-text regions: The proposed method also exploits line segments as well as text-lines in document images by introducing $f_{line}(\Theta)$.

Also, there is a trivial solution in the cost function (the focal length f is very small or large), the proposed method adds the regularization term $f_{regular}(f)$ imposing a constraint for the scale of focal length f. This regularization term is designed as

$$f_{regular}(f) = \left(\frac{\max(a, f)}{\min(a, f)} - 1 \right)^2, \tag{3.5}$$

where $a = \max(w, h)$ is similar to the method in [51,52], w and h are the width and



(a) (b) (c)

Figure 3.3: Comparison of the proposed method with a conventional text-line based method [1]. (a) Input image, (b) Result of [1]. Distortions on text regions are largely removed, however, new distortions are introduced in other regions. (c) Result of the proposed method. The proposed method exploits the properties of (red) line segments as well as (green) text-lines, and can remove overall distortions.

height of the input image, respectively.

3.1.3 Line segment properties and cost function

For the design of $f_{line}(\Theta)$, the proposed method first extracts line segments in given images by using Line Segment Detector (LSD) in [2].

In order to extract the line segments robustly (remove noise), the proposed method removes the line segments whose length are less than the mean size of texts. The mean size of texts is determined as: The Connected Components corresponding to the text components are extracted and approximated to ellipses in [50]. The text component size is determined by the major axes length of ellipse corresponding to

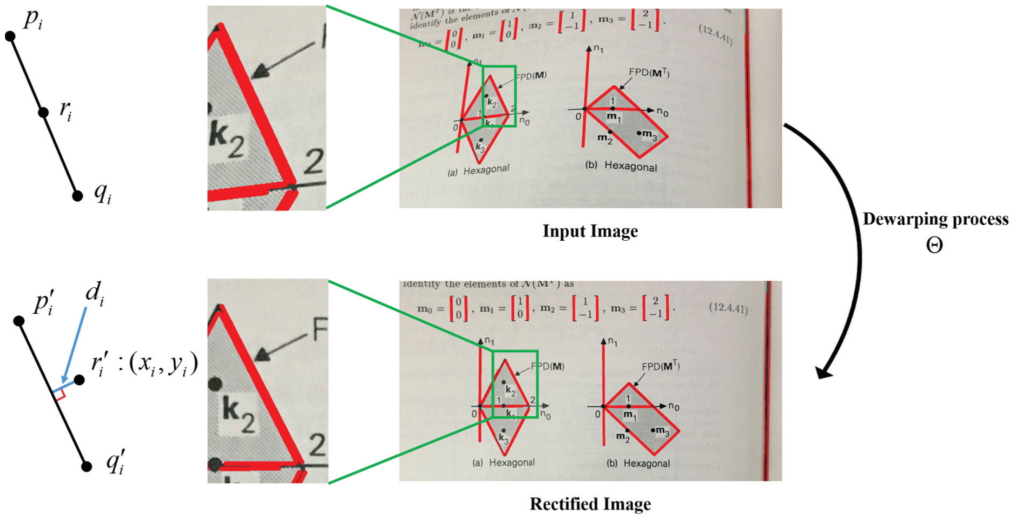


Figure 3.4: Illustration of line segment, its center point and straightness property.

the text component.

Then, based on two properties of line segments, the line segment term $f_{line}(\Theta)$ is designed as

$$f_{line}(\Theta) = \lambda_2 f_{str}(\Theta) + \lambda_3 f_{align}(\Theta), \tag{3.6}$$

where $f_{str}(\Theta)$ reflects the straightness property of line segments, and $f_{align}(\Theta)$ reflects the alignments of line segments.

First, the straightness property describes the line segments extracted in curved document image, lines on the curved document surface become still straight in the well-rectified domain (Although the lines extracted in the well-rectified image can be curved in the curved document surface). To be precise, as shown in Fig. 3.4, the proposed method denotes the end points and their center point of the i-th line segment (in the original image) as p_i, q_i and r_i, and their transformed points by the dewarping process (using Θ) as p'_i, q'_i and r'_i. Then, by the straightness property of line segments, r'_i is on the connection line between the transformed end points

p'_i and q'_i. Then, based on this straightness property of line segments, the proposed method defines the term as

$$f_{str}(\Theta) = \sum_i d_i^2 = \sum_i \frac{(a_i x_i + b_i y_i + c_i)^2}{a_i^2 + b_i^2}, \tag{3.7}$$

where d_i is the distance between the transformed center point r'_i and the transformed connection line between p'_i and q'_i. a_i, b_i and c_i are the coefficients of connection line equation between p'_i and q'_i, and (x_i, y_i) is the position of transformed center point r'_i.

As transformed line segments in well-rectified image are still straight, this term becomes small. Since the straightness property is always satisfied with all plane to plane mapping, it is not a significant constraint in rectification considering only camera view (such as homography). However the proposed method considers page curve as well as camera view in rectification process, then this property becomes an efficient constraint that prevents lines from being curved.

Then, based on the observation that the majority of line segments are horizontally or vertically aligned in the rectified images, the proposed method defines the term as

$$f_{align}(\Theta) = \sum_i \min\left(\cos^2\theta_i, \sin^2\theta_i\right), \tag{3.8}$$

where θ_i is the angle of the transformed i-th line segment (when rectified with the current parameters Θ) as illustrated in Fig. 3.5. θ_i is defined as the orientation of a line segment connecting p'_i and q'_i.

As line segments are aligned in either vertical or horizontal directions, this term becomes small as shown in Fig. 3.6. Although there are outliers (line segments having arbitrary orientations), (3.8) minimizes these effects by using bounded penalty

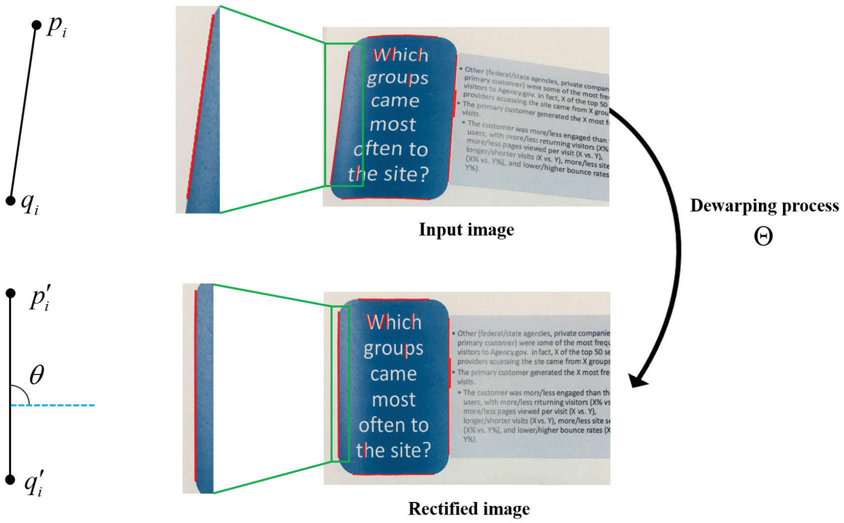
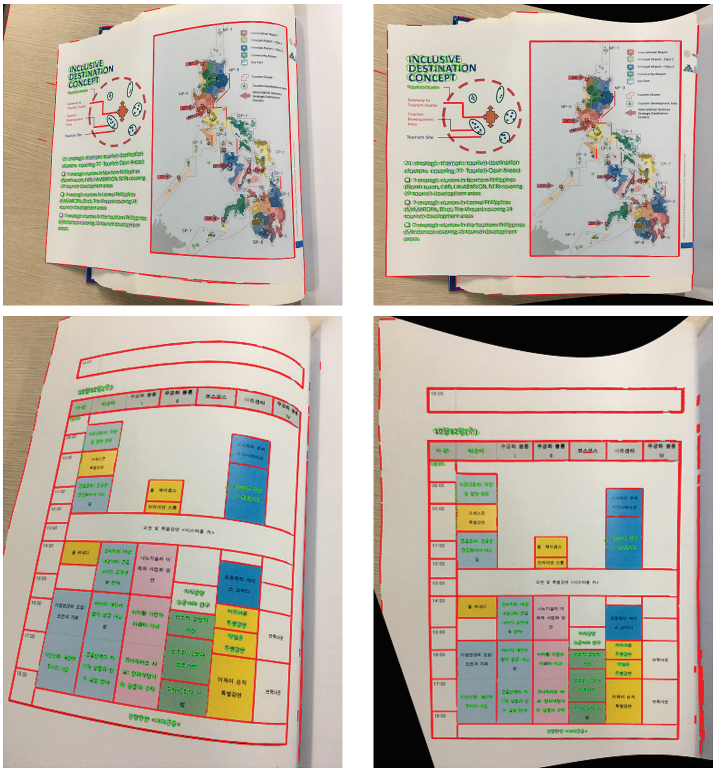


Figure 3.5: Illustration of line segment, its angle and alignment property.

functions ($\cos^2 \theta_i \leq 1$, $\sin^2 \theta_i \leq 1$). Also, the proposed method develops an optimization step that alleviates the outlier problem, this process is mentioned in next section.

In order to confirm the effectiveness of two line segment properties, the proposed method compares the rectified images by minimizing the three cost functions including only the straightness term (when $\lambda_3 = 0$), only the alignment term (when $\lambda_2 = 0$) and both two terms, as shown in Fig. 3.7. When the cost function includes only the straightness term, there are no distortions that bend straight lines (make straight lines be curved by false estimation of page curve), however the rectified image is still not well-rectified from perspective distortion. By comparison, when the image are rectified by the only alignment term, the rectified image are well aligned (with correct estimation of camera pose), however it sometimes causes distortions that make straight lines be curved by false estimation of page curve. By using both two terms, the rectified image are well rectified with the correct estimation of both

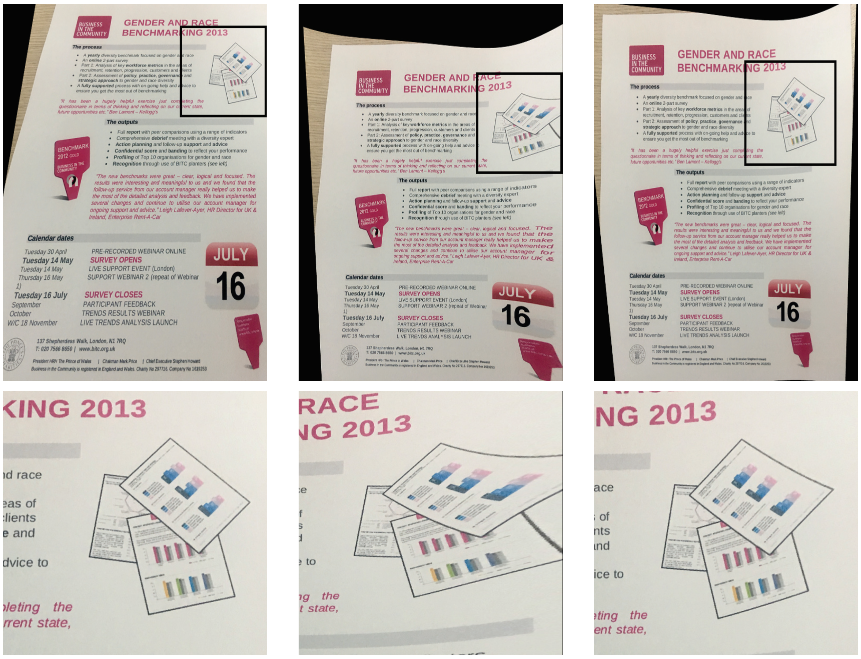


(a) (b)

Figure 3.6: Text-lines and line segments in camera-captured images and rectified images. Text components are represented as (green) ellipses and line segments are (red) lines. For example, the line segment alignment terms on bottom row are 84.3 and 0.4, respectively.

page curve and camera pose (and focal length).

In summary, the proposed cost function removes the overall distortions including text and non-text regions. As shown in Fig. 3.7, Text-line based term f_{text} removes the only distortions on text regions, however there are some distortions on non-text regions. The line segment straightness term f_{str} prevents the non-text regions in the rectified images from wrinkling (mapping straight line segments to straight line segments). Also, the alignments of line segments are more powerful assumption, then the line segment alignment term f_{align} removes all distortions from page curl and camera

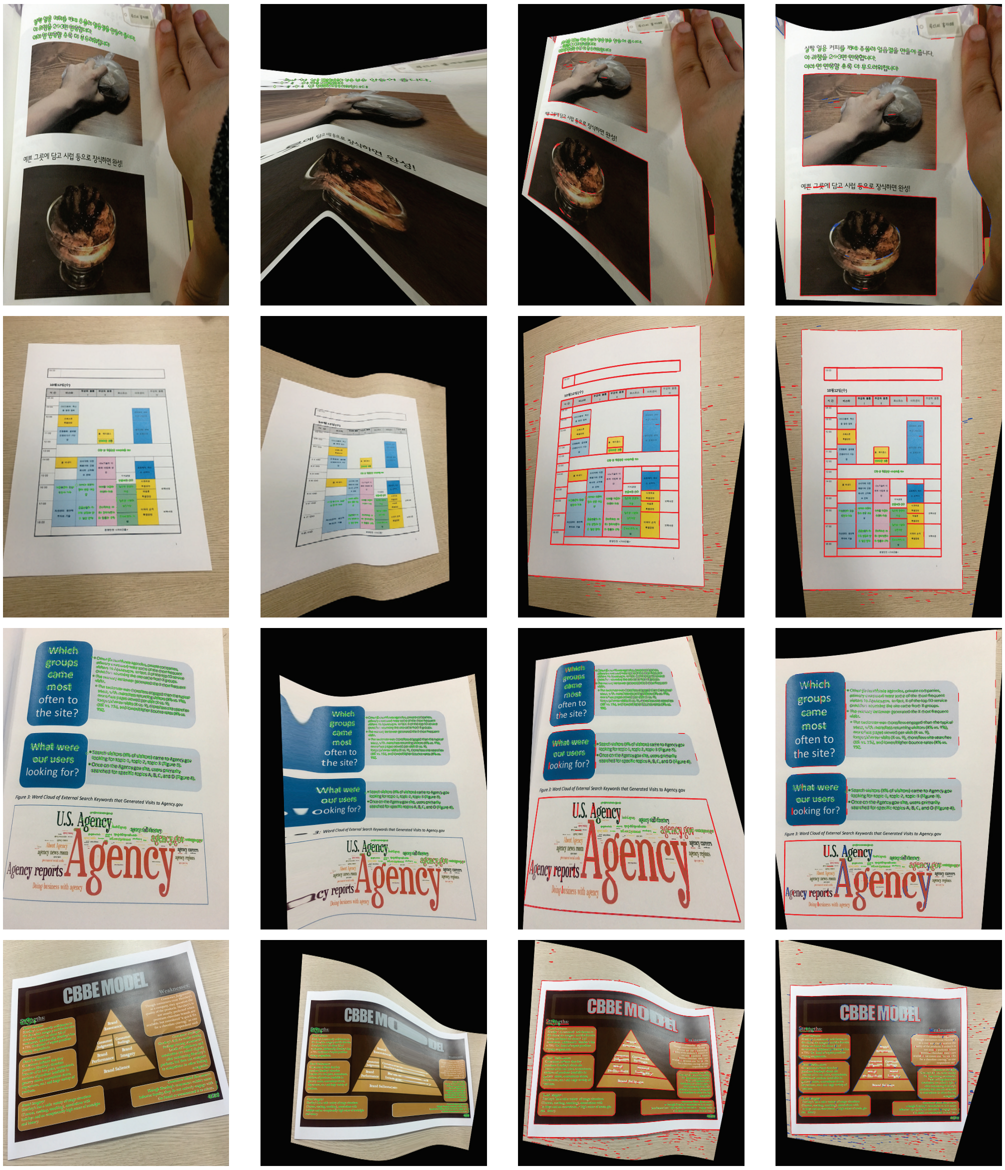


(a) (b) (c)

Figure 3.7: Rectified results by minimizing three cost functions. The first row: rectified images, the second row: the expanded images of first row. (a) Rectified images with the straightness property (sets $\lambda_3 = 0$), (b) Rectified image with the align property (sets $\lambda_2 = 0$), (c) Rectified images by minimizing the proposed cost function.

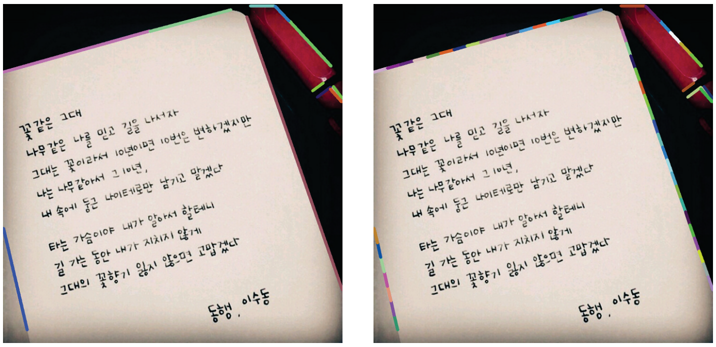
view. However, there are outliers (line segments having arbitrary orientations), the proposed method develops an optimization step that alleviates the outlier problem and the alignment term f_{align} uses only inlier (having vertical/horizontal direction) line segments.

Also, the longer line segments are more informative in rectification transformation, the proposed alignment term does not consider it. In order to give more weight to longer line segments, the proposed method performs pre-processing that divides line segment into several smaller line segments: The i-th line segment l_i is divided into n_i line segments $\{l_i^1, l_i^2, ..., l_i^{n_i}\}$, whose length are same with a length threshold t_l. Then, the number of divided line segments n_i is proportional to the length of line



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

Figure 3.8: Rectified results by minimizing the three cost functions. (a) Input images and (green) text components. (b) Rectified image by minimizing the text-line based term f_{text}, (c) Rectified image by minimizing the text-line based term f_{text} and line segment straightness term f_{str}, and (red) line segments, (d) Rectified image by minimizing the cost function f_{cost}, (red) horizontal/vertical and (blue) non horizontal/vertical line segments.



꽃같은 그대

나무같은 나를 믿고 길을 나서자
그대는 꽃이라서 10년이면 10번은 변하겠지만
나는 나무같아서 그 10년,
내 속에 둥근 나이테로만 남기고 말겠다

타는 가슴이야 내가 알아서 할테니
길 가는 동안 내가 지치지 않게
그대의 꽃향기 잃지 않으면 고맙겠다

동행, 이수동

(a) (b)

Figure 3.9: Pre-processing of line segment extraction. (a) A result of line segment extraction by [2], (b) A pre-processing result. All line segments are divided into some line segments, the number of the divided line segments are proportional to the length of line segment.

segment l_i. Consequently, the longer lines are divided into more line segments, its direction is more considered in the alignment term.

3.2 Outlier removal and optimization

Although the optimization method used in [1] assumes that there are no outliers (or their effects are not critical), the direct optimization of $f_{cost}(\Theta)$ may yield poorly rectified results as shown in Fig. 3.10-(b), due to outliers (false-positive of text line detection and line segments having arbitrary orientations). For the outlier removal, the proposed method designs an iterative scheme. At each step, the proposed method removes outliers and minimizes the cost function with updated inliers, that are horizontal/vertical line segments and horizontal straight text-lines. To be precise, an updated inlier set at the $(j+1)$-th iteration is defined as

$$\begin{aligned} T_{j+1} &= \{t | t \in t_j, f_{text-str}(l) < \rho_j\}, \\ L_{j+1} &= \{l | l \in L_j, f_{align}(l) < \tau_j\}, \end{aligned} \tag{3.9}$$

where $f_{text-str}$ is the term reflecting the property for horizontal text-lines in text-line based term f_{text}. It becomes small when the text-lines are horizontal. T_j and L_j are inlier sets of text components and line segments at the j-th iteration. Since these terms reflects the properties of text-lines and line segment alignments, the proposed method detects outliers whose the cost terms are more than threshold. This iteration is repeated until the number of inliers becomes stable. After extensive experiments, the proposed method confirms that the number of iteration is usually $1 \sim 3$.

The proposed method computes the jacobian matrix of the cost function, then the cost function can be minimized via the Levenberg-Marquardt (LM) algorithm [14,15].

3.2.1 Jacobian matrix of the proposed cost function

Derivatives of the text-line based term

In [1], for the rectification parameters $(\{a_m\}_{m=0}^{M}, \mathbf{R})$, the derivatives of the text-line based term f_{text} are computed, the proposed method uses them. In addition, since the proposed method adds camera focal length f in the rectification parameters, then computes the derivative of the text-line based term f_{text} with respect to focal length f.

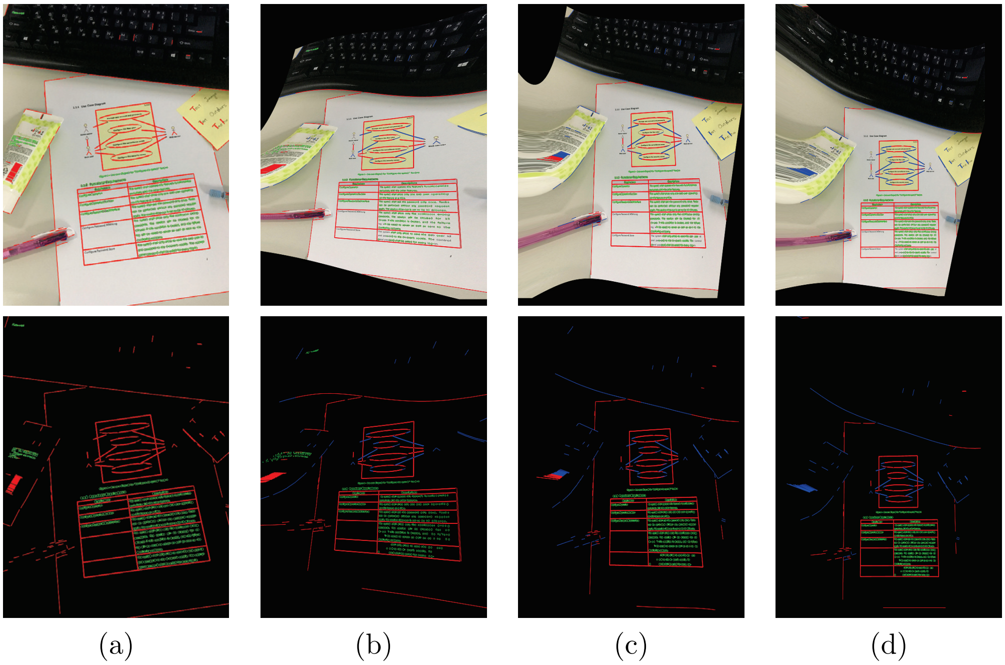


Figure 3.10: The proposed iterative scheme. (a) An input image, (b) Result after the first iteration, (c) Result after the second iteration, (d) Result after the third iteration. At each step, (blue lines) outliers are removed and (red lines, green texts) are updated.

First, the equation of camera model in (3.1) can be represented as:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \mathbf{R}^{\top} \left[k\boldsymbol{p} - \boldsymbol{t} \right], \tag{3.10}$$

where $\boldsymbol{p} = ((\alpha - c_x), (\beta - c_y), -f)^{\top}$ and $\boldsymbol{t} = (0, 0, f)^{\top}$. Then, the proposed method differentiates both sides of (3.3) and (3.10) with respect to f:

$$g'(x)\frac{\partial x}{\partial f} = \frac{\partial z}{\partial f}, \tag{3.11}$$

$$\frac{\partial}{\partial f}\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \mathbf{R}^\top \left[\frac{\partial k}{\partial f}\boldsymbol{p} - (k+1)\begin{pmatrix} 0 \\ 0 \\ f \end{pmatrix} \right]. \tag{3.12}$$

Then, the proposed method can get

$$\frac{\partial k}{\partial f} = (k+1)\frac{(r_{2,2} - g'(x)r_{0,2})}{(\boldsymbol{r_3}^\top \boldsymbol{p} - g'(x)\boldsymbol{r_1}^\top \boldsymbol{p})}, \tag{3.13}$$

where $r_{i,j}$ is the i-th row and j-th column element of the matrix $\mathbf{R}^\top$, and $\boldsymbol{r_i}$ is the i-th column vector of the matrix $\mathbf{R}^\top$.

Then, the derivative of text-line based term $\frac{\partial}{\partial f} f_{text}$ can be computed using chain-rule, this process is mentioned in [1].

Derivatives of line segment terms

The text-line based term of the cost function is computed in the curved document domain (x, y, z), however the line segment terms are computed in the rectified document domain (u, v). In order to compute the derivatives on the rectified document domain, the proposed method first uses a simple approximation in (3.3) using the Simpson‘s rule [53]:

$$\begin{aligned} u &= \int_0^x \sqrt{1 + g'(t)^2} dt \\ &= \frac{x}{6}\left[\sqrt{1 + g'(0)^2} + 4 * \sqrt{1 + g'(\frac{x}{2})^2} + \sqrt{1 + g'(x)^2} \right], \\ v &= y. \end{aligned} \tag{3.14}$$

Then, the proposed method can compute the derivatives of the point (u, v) with respect to parameter t using chain-rule:

$$\begin{aligned}
\frac{\partial u}{\partial t} &= \frac{\partial u}{\partial x}\frac{\partial x}{\partial t}, \\
&= \frac{1}{6}\left[\sqrt{1+g'(0)^2} + 4 * \sqrt{1+g'(\frac{x}{2})^2} + \sqrt{1+g'(x)^2}\right]\frac{\partial x}{\partial t} \\
&+ \frac{x}{6}\left[2 * \frac{g'(\frac{x}{2})g''(\frac{x}{2})}{\sqrt{1+g'(\frac{x}{2})^2}} + \frac{g'(x)g''(x)}{\sqrt{1+g'(x)^2}}\right]\frac{\partial x}{\partial t} \\
\frac{\partial v}{\partial t} &= \frac{\partial y}{\partial t}.
\end{aligned} \tag{3.15}$$

Then, $\frac{\partial u}{\partial t}$ and $\frac{\partial v}{\partial t}$ can be computed using $\frac{\partial x}{\partial t}$ and $\frac{\partial y}{\partial t}$.

Also, the $\min(\cdot,\cdot)$ function in the alignment term (3.8) is not differentiable at some points, therefore the proposed method uses a simple approximation:

$$\frac{\partial}{\partial t}(\min(f(\cdot), g(\cdot))) = \begin{cases} \frac{\partial f(\cdot)}{\partial t}, & \text{if } f(\cdot) \leq g(\cdot) \\ \frac{\partial g(\cdot)}{\partial t}, & \text{otherwise.} \end{cases} \tag{3.16}$$

Derivatives of the regularization term

The derivatives of the regularization term with respect to the camera focal length f is computed as:

$$\frac{\partial}{\partial t}(f_{regular}) = \begin{cases} \frac{1}{a}, & \text{if } a < f \\ -\frac{a}{f^2}, & \text{if } f < a \\ 0, & \text{otherwise.} \end{cases} \tag{3.17}$$

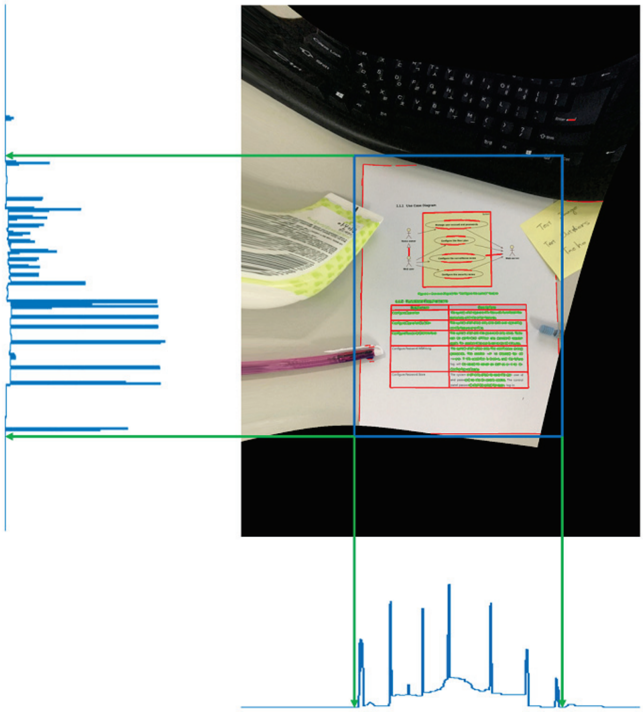


Figure 3.11: Document region detection. A blue rectangle is a result of the document region detection.

3.3 Document region detection and dewarping

In this section, the proposed method detects the document region. Documents contain many text and non-text (such as tables and figures) regions, and non-text regions generally contain many horizontal/vertical lines. In addition, the border lines of document page also have horizontal/vertical direction. Based on this observation, the proposed method determines the document region using a projection profile method. The proposed method computes projection profiles on horizontal/vertical directions. Then the proposed method sets the border lines of document region whose projection profiles more than threshold values, while including updated inlier text components, as shown in Fig. 3.11.

Lastly, the proposed method takes the rendering of the distorted image using the estimated rectification parameters. In this rendering process, computation of rectified pixel positions corresponding to the whole input pixels is somewhat inefficient in the aspect of time consuming. Then, the proposed method computes the positions of only rectified pixels inside corresponding to the document region, then take rendering of the distorted image.

3.4 Experimental results

In the experiments, the proposed method sets the weight $\lambda_1 = 100$, and λ_2, λ_3 in (3.4) so that they are proportional to $\frac{N_{text}}{N_{line}}$, where N_{text} and N_{line} are the numbers of text-lines and line segments, respectively. Also, the length threshold t_l for preprocessing is proportional to the mean size of texts. Also, the proposed method sets the initial value ($j = 1$) of threshold τ_j for outlier removal to 0.01, and this threshold becomes half as increasing iteration number j. The proposed method is implemented with C++ and the proposed implementation takes $4 \sim 6.3$ (s) for the rectification of an image (4000×3000) on Intel(R) i5(TM) CPU(3.40GHz). To be precise, Text-line extraction time takes 2.8 (s), line segment extraction time takes 1.1 (s), optimization of cost function time takes 0.8 (s), and rendering process takes about 1.6 (s). Many of the processing time is spent on the text line extraction and rendering process.

For the evaluation, the proposed method conducted experiments on two types of datasets: text-abundant images and non conventional document images (i.e., not text-abundant cases).

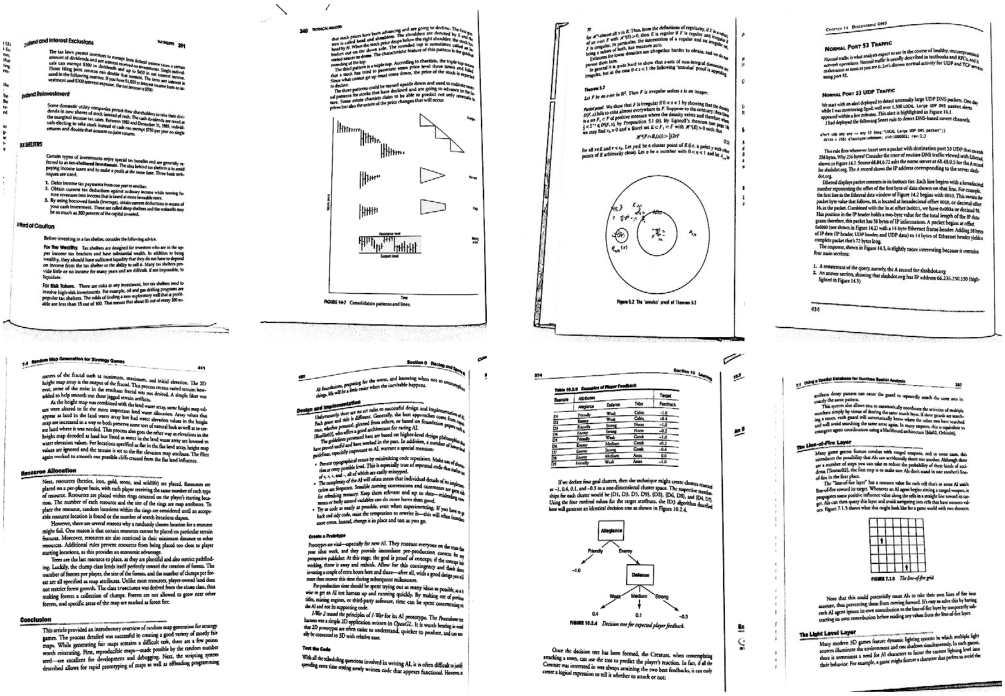


Figure 3.12: Images of CBDAR2007 dataset. They are text-abundant document images captured by a digital camera.

3.4.1 Experimental results on text-abundant document images

The proposed method first conducted experiments on text-abundant images, CBDAR2007 dewarping contest dataset [54]. The CBDAR2007 dewarping contest dataset consists of 102 binarized document images as shown in the in Fig. 3.12.

This dissertation first evaluates the proposed method on CBDAR2007 dataset and compares the performance with the conventional methods [1, 3, 29, 55, 56] in terms of OCR accuracy. To be precise, the accuracy is defined as

$$\text{accuracy}(R, G) = 1 - \frac{L(R, G)}{\max(\#R, \#G)}, \tag{3.18}$$

where R is a recognition result, G is the ground truth, $\#(\cdot)$ is the number of

Table 3.1: OCR performance on CBDAR2007 dewarping contest dataset

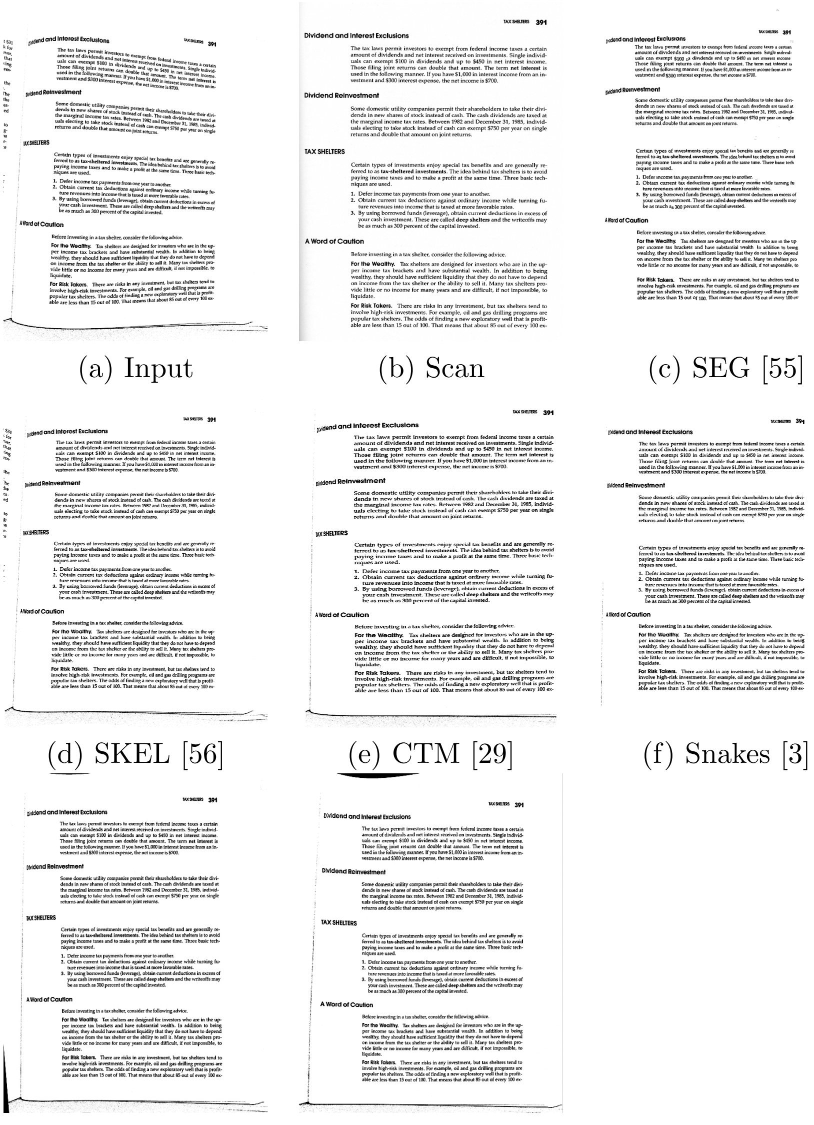
	Original	SEG [55]	SKEL [56]	CTM [29]	Snakes [3]	Kim [1]	Proposed
mean accuracy	62.47	89.47	93.17	96.22	96.47	97.42	**97.82**

characters in the string, and $L(x, y)$ means the Levenshtein distance between two strings [57]. The distance is defined as the minimum number of character edits (insertion, deletion, and substitution) to transform one string to the other. For the OCR, the proposed method uses the Google tesseract OCR engine [7]. Experimental results are summarized in Table. 3.1. Since samples in CBDAR2007 dewarping contest dataset are text-abundant images (having single columns), the conventional text-line based method [1] showed good performance. However, as can be seen, the proposed method can also handle all these cases and shows improved accuracy (probably due to the proposed method considering outliers and estimation of focal length).

Some experimental results are shown in Fig. 3.13, Fig. 3.14 and 3.15. Since the binding lines of book and the border lines of figures and papers are horizontal/vertical aligned, the results of the proposed method has more less distortions on both text and non-text regions. Also, some experimental results in Fig. 3.16 and Fig. 3.17 shows that proposed outlier removal scheme works well. Although the inputs have miss-detected texts and non horizontal/vertical lines, the proposed method removes these outliers and rectifies inputs using updated inlier features.

3.4.2 Experimental results on non conventional document images

In order to consist of non conventional document images (i.e., not text-abundant cases), the proposed method collected 100 images having various layouts (e.g., three



TAX SHELTERS 391

Dividend and Interest Exclusions

The tax laws permit investors to exempt from federal income taxes a certain amount of dividends and net interest received on investments. Single individuals can exempt $100 in dividends and up to $450 in net interest income. Those filing joint returns can double that amount. The term **net interest** is used in the following manner. If you have $1,000 in interest income from an investment and $300 interest expense, the net income is $700.

Dividend Reinvestment

Some domestic utility companies permit their shareholders to take their dividends in new shares of stock instead of cash. The cash dividends are taxed at the marginal income tax rates. Between 1982 and December 31, 1985, individuals electing to take stock instead of cash can exempt $750 per year on single returns and double that amount on joint returns.

TAX SHELTERS

Certain types of investments enjoy special tax benefits and are generally referred to as **tax-sheltered investments**. The idea behind tax shelters is to avoid paying income taxes and to make a profit at the same time. Three basic techniques are used.

1. Defer income tax payments from one year to another.
2. Obtain current tax deductions against ordinary income while turning future revenues into income that is taxed at more favorable rates.
3. By using borrowed funds (leverage), obtain current deductions in excess of your cash investment. These are called **deep shelters** and the writeoffs may be as much as 300 percent of the capital invested.

A Word of Caution

Before investing in a tax shelter, consider the following advice.

For the Wealthy. Tax shelters are designed for investors who are in the upper income tax brackets and have substantial wealth. In addition to being wealthy, they should have sufficient liquidity that they do not have to depend on income from the tax shelter or the ability to sell it. Many tax shelters provide little or no income for many years and are difficult, if not impossible, to liquidate.

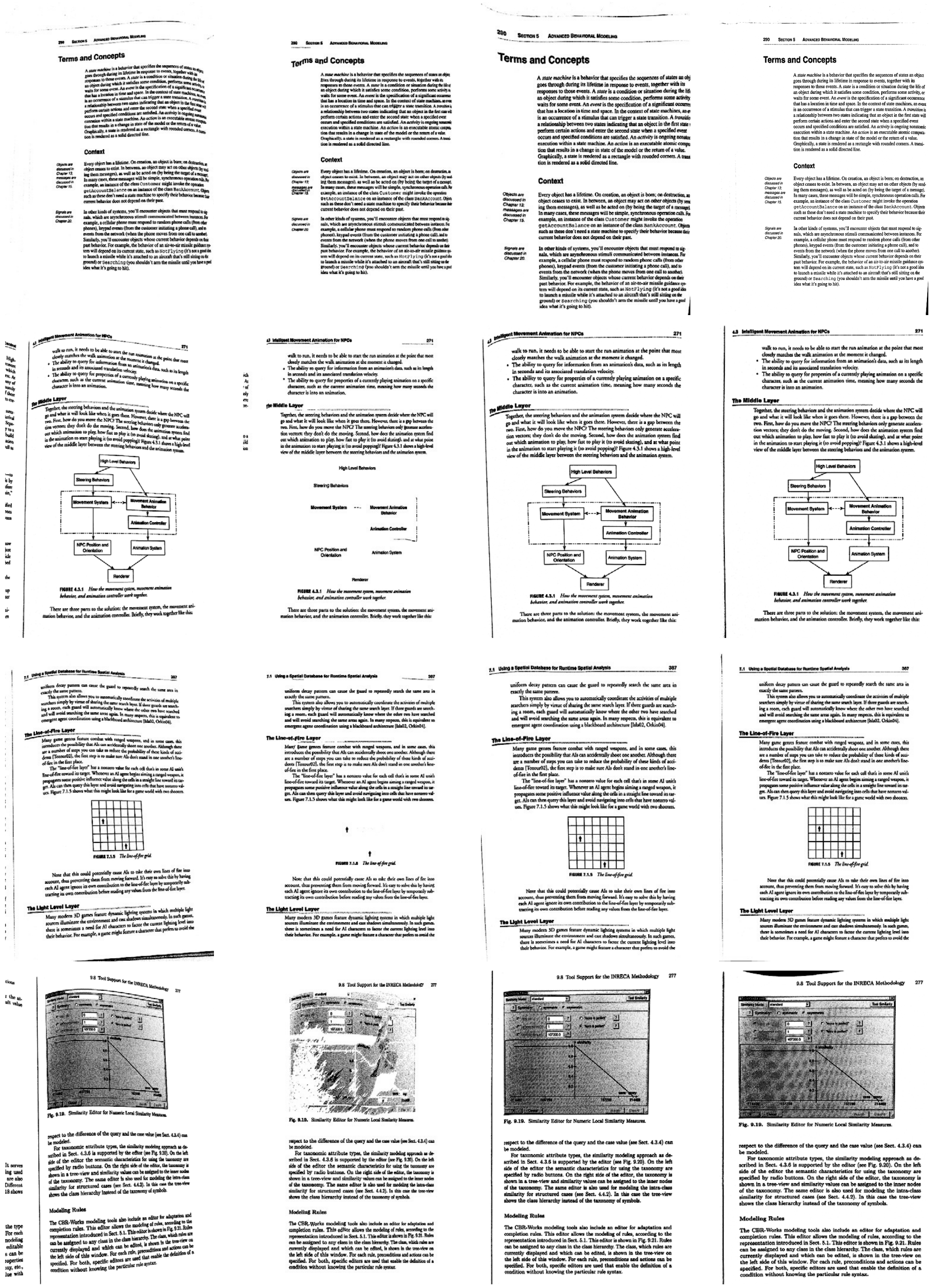
For Risk Takers. There are risks in any investment, but tax shelters tend to involve high-risk investments. For example, oil and gas drilling programs are popular tax shelters. The odds of finding a new exploratory well that is profitable are less than 15 out of 100. That means that about 85 out of every 100 ex-

(a) Input (b) Scan (c) SEG [55]

(d) SKEL [56] (e) CTM [29] (f) Snakes [3]

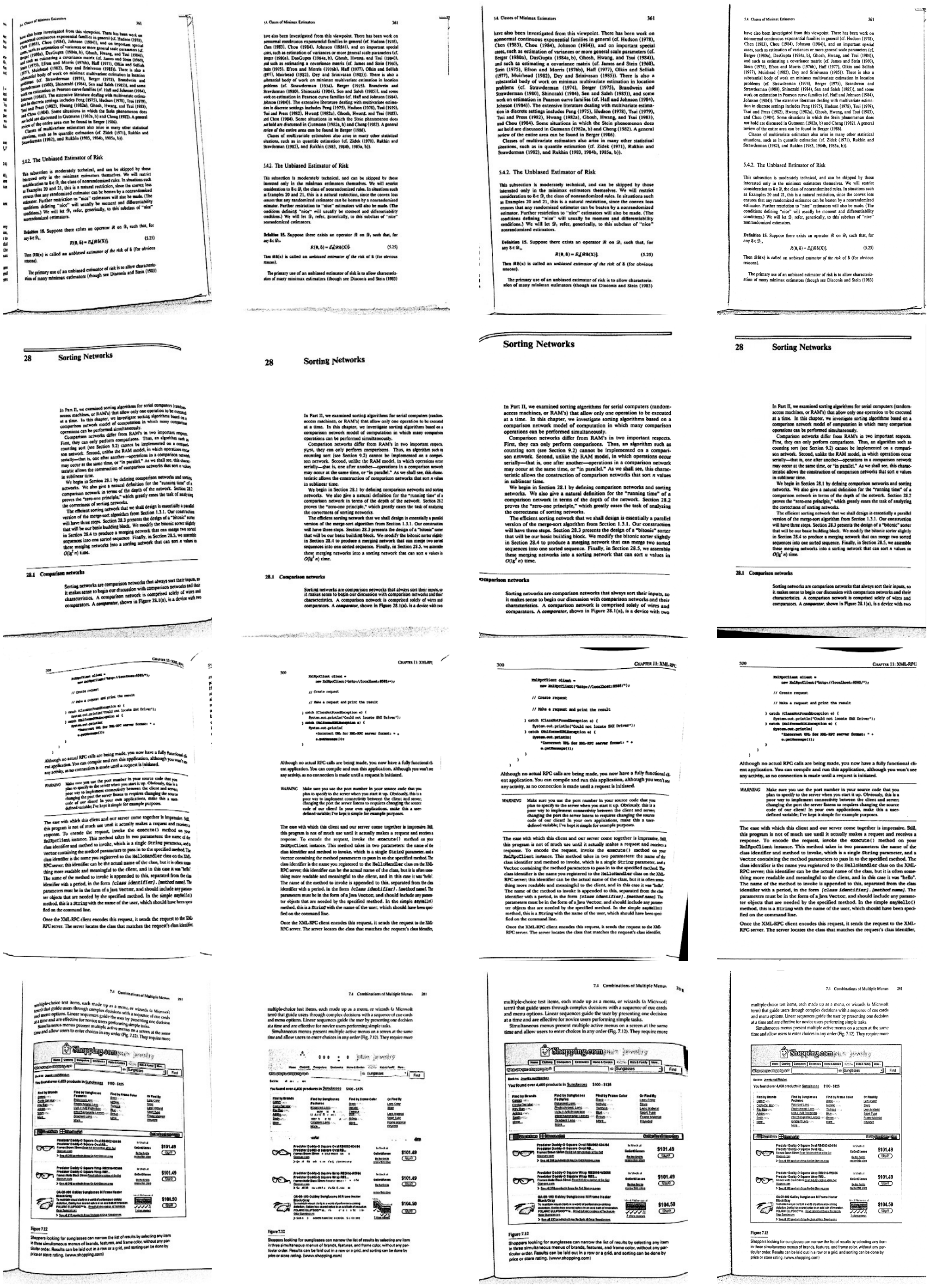
(g) Kim [1] (h) Proposed

Figure 3.13: Results of CBDAR2007 dataset.



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

Figure 3.14: Results of CBDAR2007 dataset, (a) Input images, (b) Results of [3], (c) Results of [1], (d) Results of the proposed method.



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

Figure 3.15: Results of CBDAR2007 dataset, (a) Input images, (b) Results of [3], (c) Results of [1], (d) Results of the proposed method.

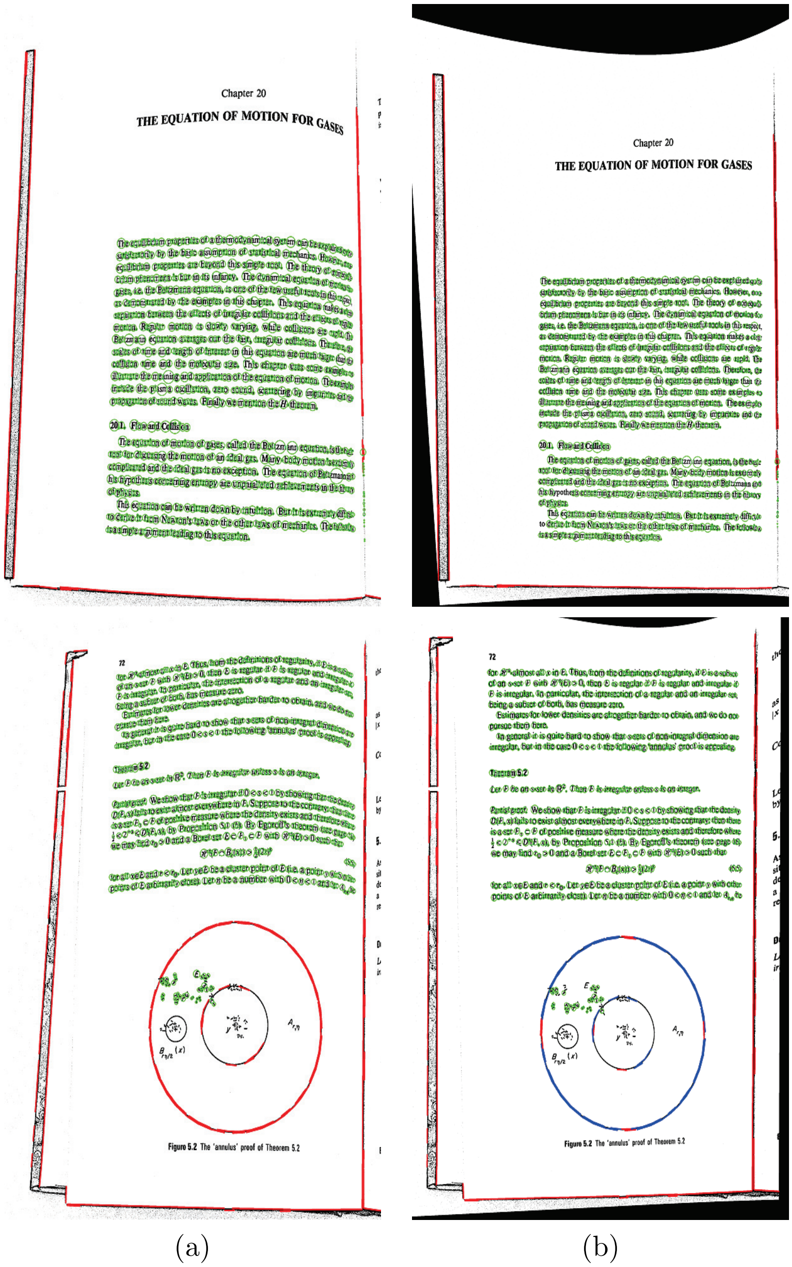


Figure 3.16: Results of CBDAR2007 dataset and illustration of features used for rectification, (a) Input images and extracted features (red lines and green text components), (b) Results of the proposed method, inlier features (red vertical/horizontal aligned lines and green text components in horizontal text-liines), and outlier blue lines that have arbitrary direction.

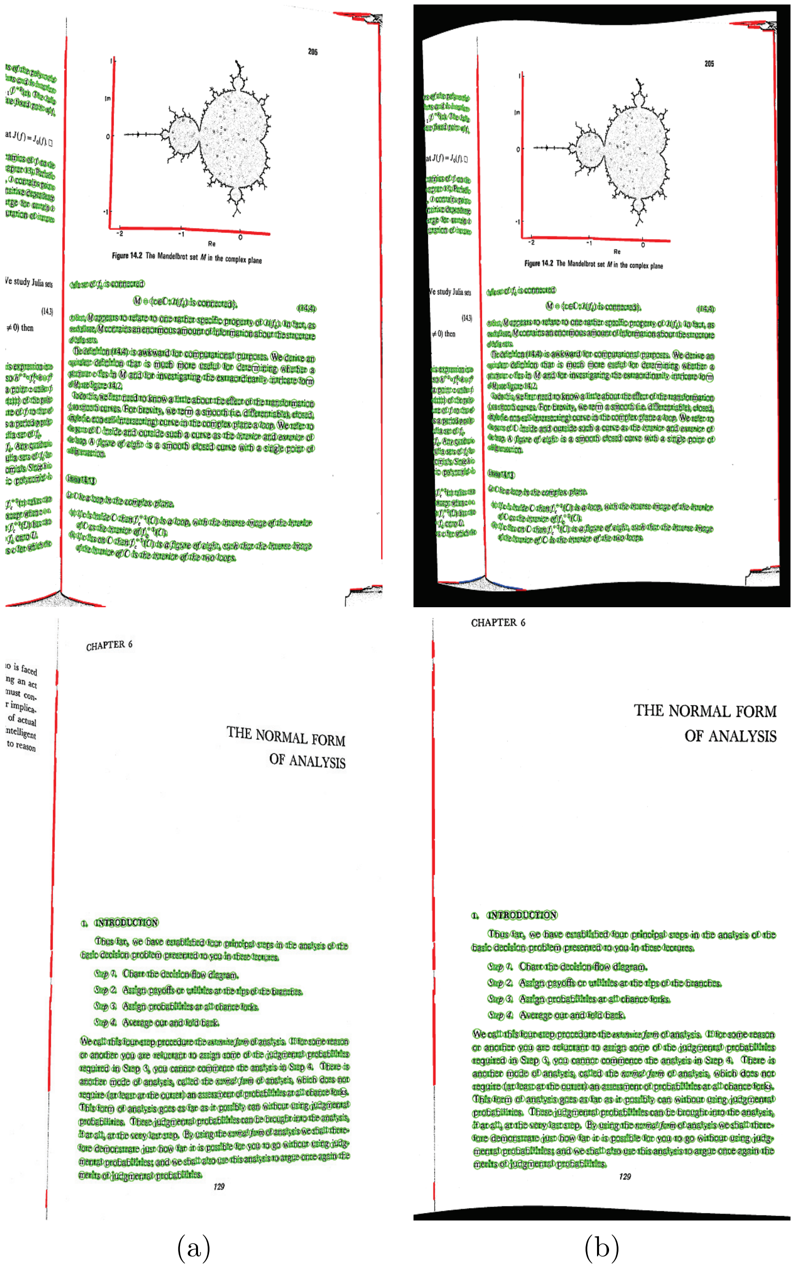


Figure 3.17: Results of CBDAR2007 dataset and illustration of features used for rectification, (a) Input images and extracted features (red lines and green text components), (b) Results of the proposed method, inlier features (red vertical/horizontal aligned lines and green text components in horizontal text-liines), and outlier blue lines that have arbitrary direction.

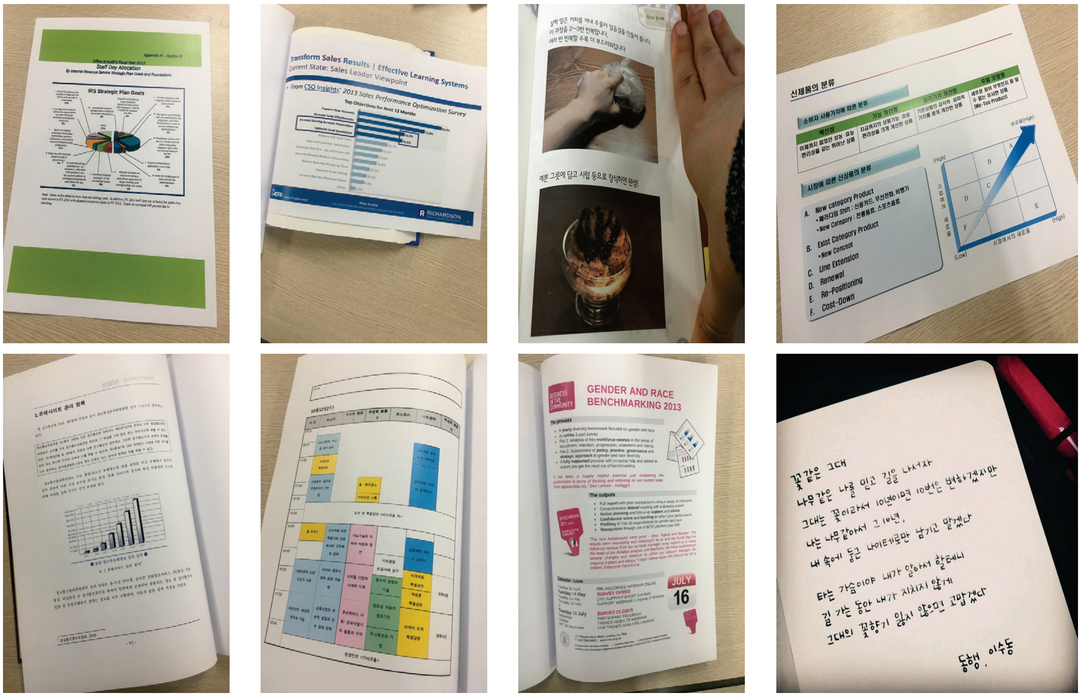


Figure 3.18: Images of the non conventional document dataset collected by this dissertation. They have various layouts.

column documents, documents containing large tables and/or figures, presentation slides, and so on) as shown in the Fig. 3.18.

For the evaluation on non conventional cases, the proposed method also conducted experiments on the dataset of the proposed method (100 images). Since the proposed method wants to evaluate the rectification performance on non-text regions, the proposed method computed the geometric quantities of rectangles in rectified results. To be precise, the proposed method uses orthogonality θ_o, diagonal

Table 3.2: Geometric measures of the proposed and conventional methods on our dataset

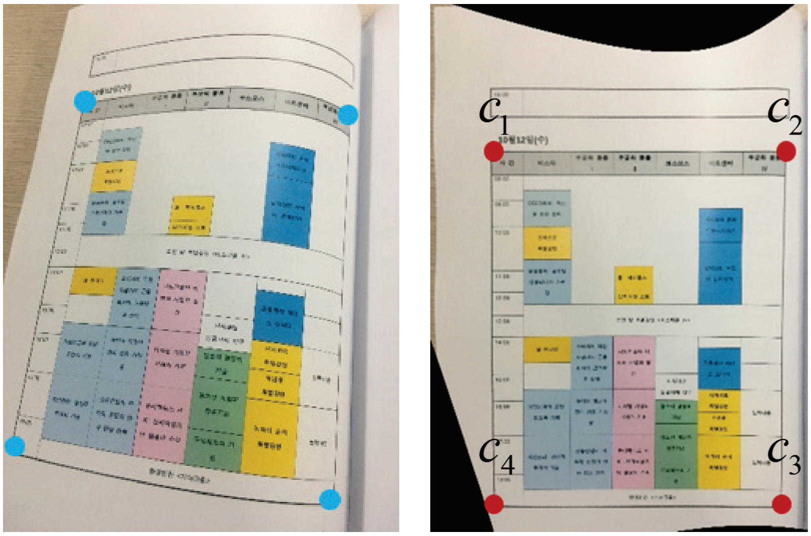
	Original	Kim [1]	Proposed
Orthogonality	24.2816	12.8471	**1.9181**
Diagonal ratio	0.0629	0.0415	**0.0089**
Vertical ratio	0.1826	0.0950	**0.0289**
Horizontal ratio	0.1427	0.0860	**0.0241**

ratio r_d, and length ratios for opposite sides r_h and r_v [52], which are defined as

$$\begin{aligned}
\theta_o &= \cos^{-1}\left(\frac{(\boldsymbol{c_1}-\boldsymbol{c_2})\cdot(\boldsymbol{c_1}-\boldsymbol{c_4})}{d(\boldsymbol{c_1},\boldsymbol{c_2})\times d(\boldsymbol{c_1},\boldsymbol{c_4})}\right), \\
r_d &= \max\left(\frac{d(\boldsymbol{c_1},\boldsymbol{c_3})}{d(\boldsymbol{c_2},\boldsymbol{c_4})},\frac{d(\boldsymbol{c_2},\boldsymbol{c_4})}{d(\boldsymbol{c_1},\boldsymbol{c_3})}\right), \\
r_h &= \max\left(\frac{d(\boldsymbol{c_1},\boldsymbol{c_4})}{d(\boldsymbol{c_2},\boldsymbol{c_3})},\frac{d(\boldsymbol{c_2},\boldsymbol{c_3})}{d(\boldsymbol{c_1},\boldsymbol{c_4})}\right), \\
r_v &= \max\left(\frac{d(\boldsymbol{c_1},\boldsymbol{c_2})}{d(\boldsymbol{c_3},\boldsymbol{c_4})},\frac{d(\boldsymbol{c_3},\boldsymbol{c_4})}{d(\boldsymbol{c_1},\boldsymbol{c_2})}\right)
\end{aligned}$$

where $\boldsymbol{c_i}$ $(i = 1, 2, 3, 4)$ is a corner point of a rectangle as shown in Fig. 3.19. Since $\theta_o = 90°$ and $r_d = r_h = r_v = 1$ in ideally rectified images, the proposed method measures the remaining geometric distortions with $|\theta_o - 90°|$, $|r_d - 1|$, $|r_h - 1|$ and $|r_v - 1|$. Experimental results are summarized in Table. 3.2. Since the executable of [1] is publicly available, it (author's implementation) is compared with the proposed method. As shown, the proposed method shows improved geometric rectification performance in terms of all measures.

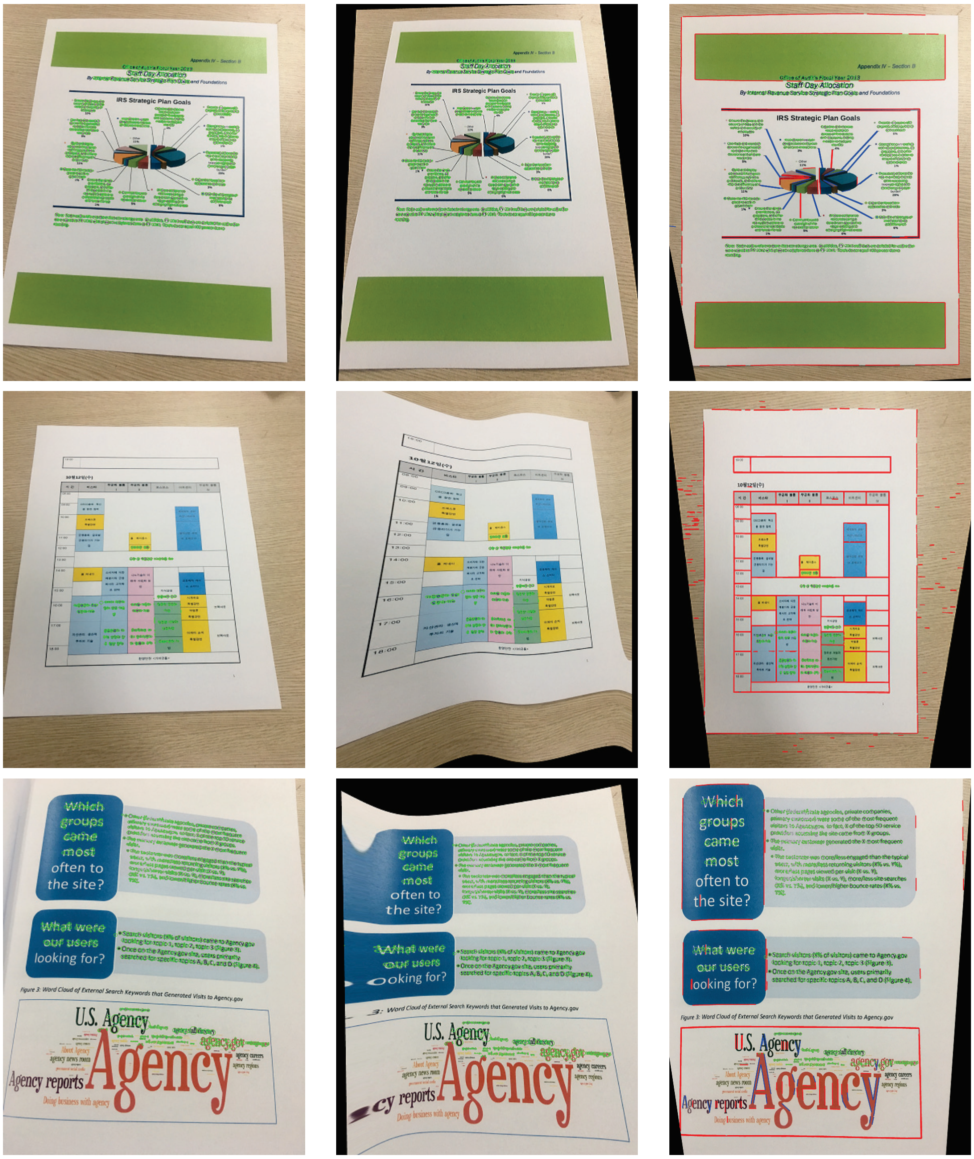
Some experimental results are shown in Fig. 3.20 and 3.21. Unlike the proposed method, the conventional text-line based method has difficulties when there are



(a) (b)

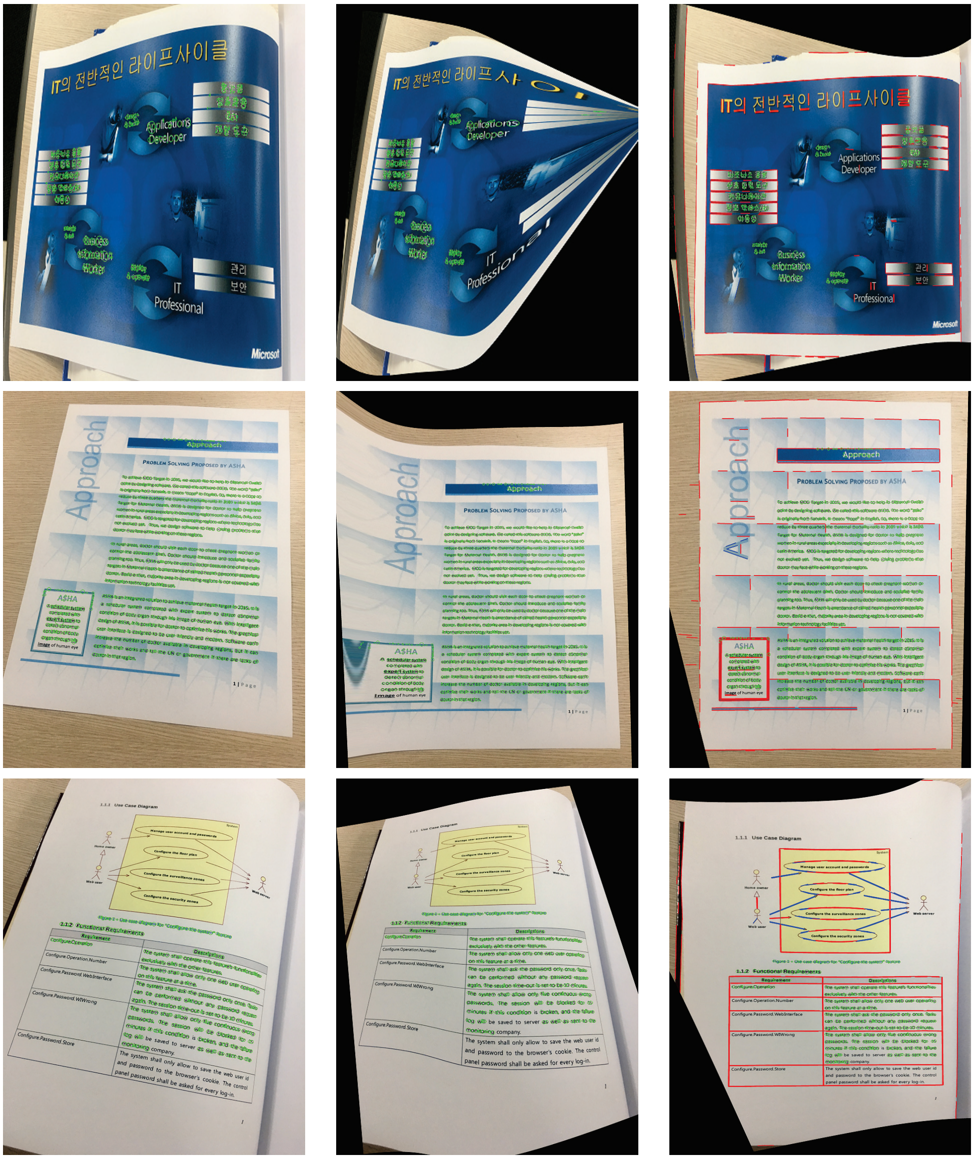
Figure 3.19: Four corner points extraction for evaluation. (a) The distorted images and manually annotated corner (blue) points, (b) The transformed images and corner (red) points.

few aligned text-lines or mis-detected text-lines (false positives). However, since the proposed method exploits line segments and removes outliers, the proposed method works robustly for a variety of inputs. Also, as shown in Fig. 3.22 and 3.23, the proposed method can handle the documents containing arbitrary graphical entities (such as circle and ellipse). The proposed method exploits text-lines as well as line segments and considers outlier line segments that have arbitrary direction. Also, the boundary of page and bookbinding regions always contain the well-aligned (horizontal or vertical) lines. Therefore, it can enough handle documents containing arbitrary graphical entities.



(a) (b) (c)

Figure 3.20: Experimental results on the proposed datasets. (a) Distorted input images and (green) text components, (b) Rectified images by the text-line based method [1], (c) Rectified images by the proposed method and (red) horizontal/vertical and (blue) non horizontal/vertical line segments.



(a) (b) (c)

Figure 3.21: Experimental results on the proposed datasets. (a) Distorted input images and (green) text components, (b) Rectified images by the text-line based method [1], (c) Rectified images by the proposed method and (red) horizontal/vertical and (blue) non horizontal/vertical line segments.

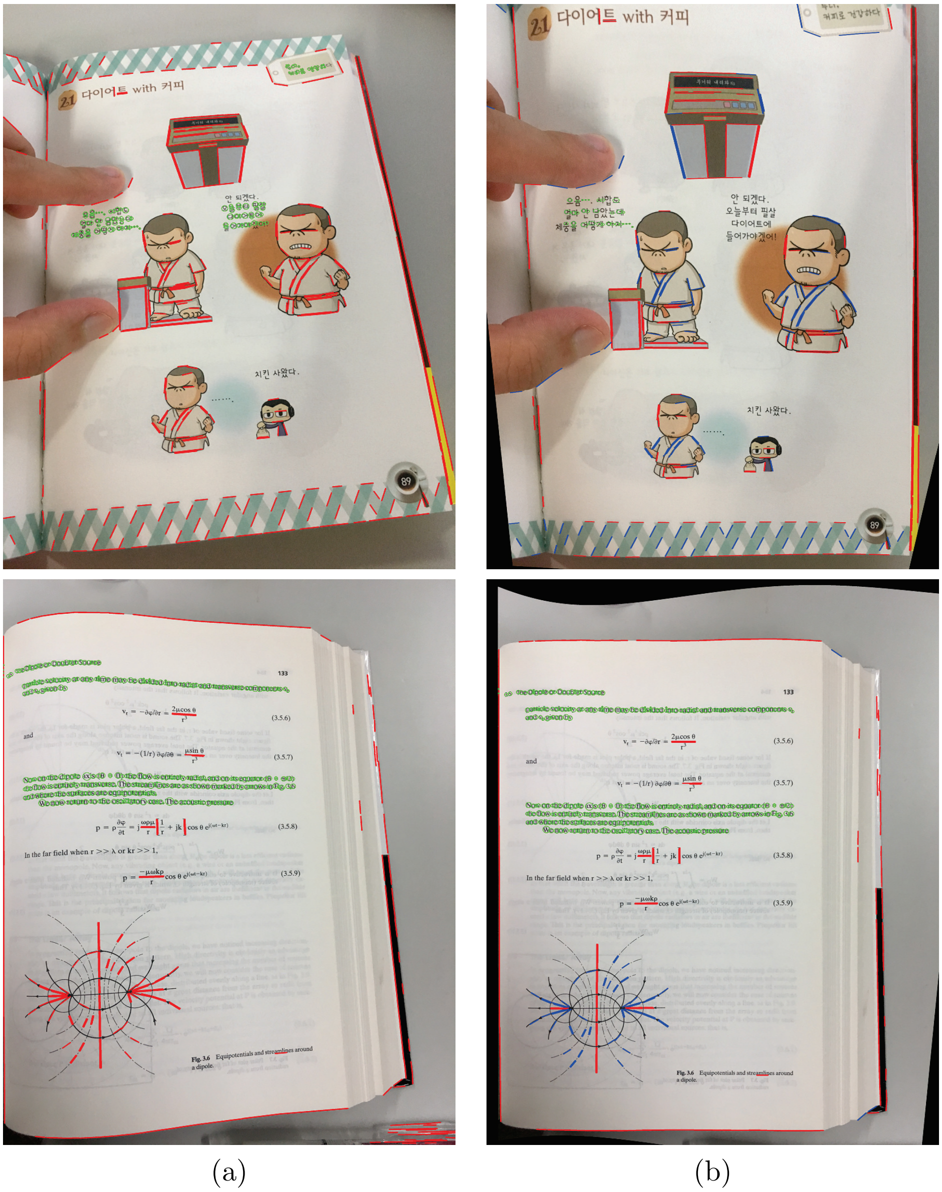
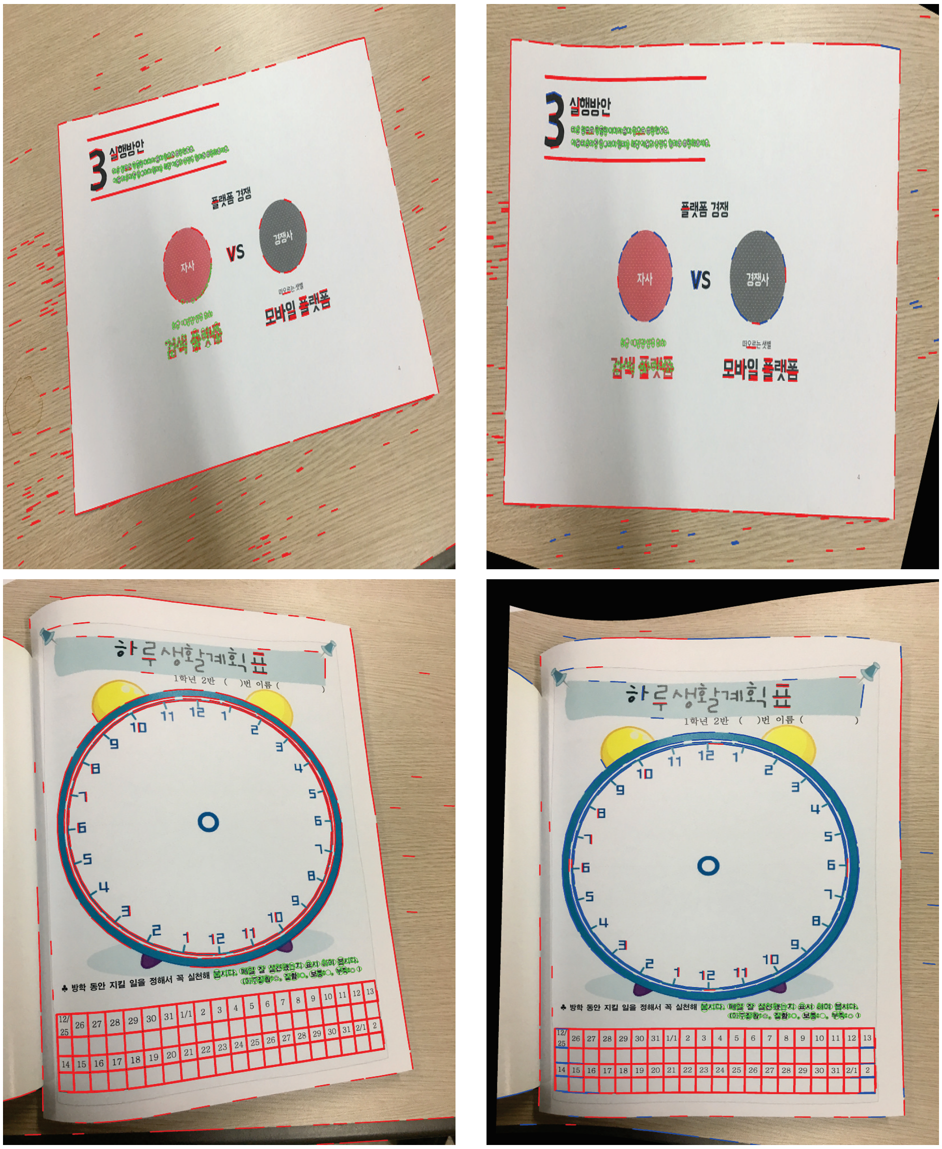


Figure 3.22: Experimental results on the proposed datasets. (a) Distorted input images and (green) text components, (b) Rectified images by the proposed method and (red) horizontal/vertical and (blue) non horizontal/vertical line segments.



(a) (b)

Figure 3.23: Experimental results on the proposed datasets. (a) Distorted input images and (green) text components, (b) Rectified images by the proposed method and (red) horizontal/vertical and (blue) non horizontal/vertical line segments.

3.5 Summary

In this chapter, this dissertation has proposed a document dewarping method exploiting the properties of line segments and text-lines. The proposed method for the document dewarping encoded the straightness and alignment properties of line segments into the proposed cost function so that the method works on both text and non-text regions. Also, the proposed method developed an iterative optimization scheme in order to handle outliers. Experimental results showed that the proposed method yields the state of the art OCR performance on text regions and visually pleasing rectified results on non-text regions.

Chapter 4

Scene text rectification

4.1 Proposed cost function for rectification

The proposed cost function for scene text image rectification is presented in this section. First, the proposed cost function reflecting glyph and alignment properties is introduced, and alignment properties and their terms are presented in detail.

4.1.1 Cost function design

Similar to conventional methods [4,39–44], the proposed method assumes that scene text is on planar surfaces and text regions are already detected and binarized [9]. Then, the transformation parameters can be modeled as the projective transformation.

For the estimation of transformation parameters τ, the proposed method devel-

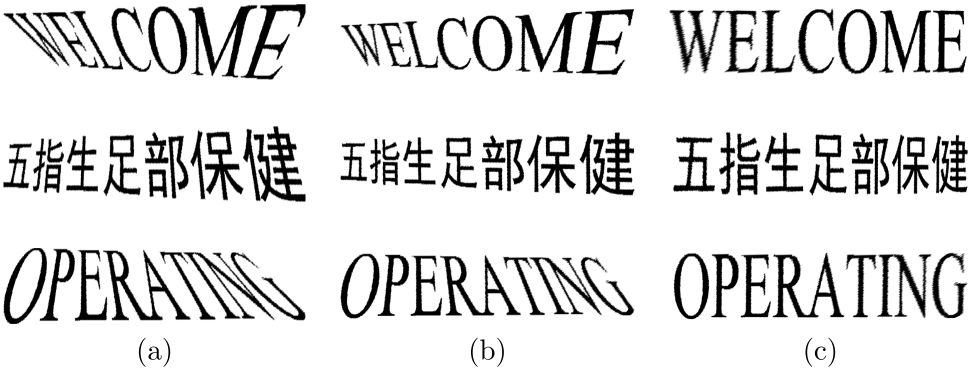
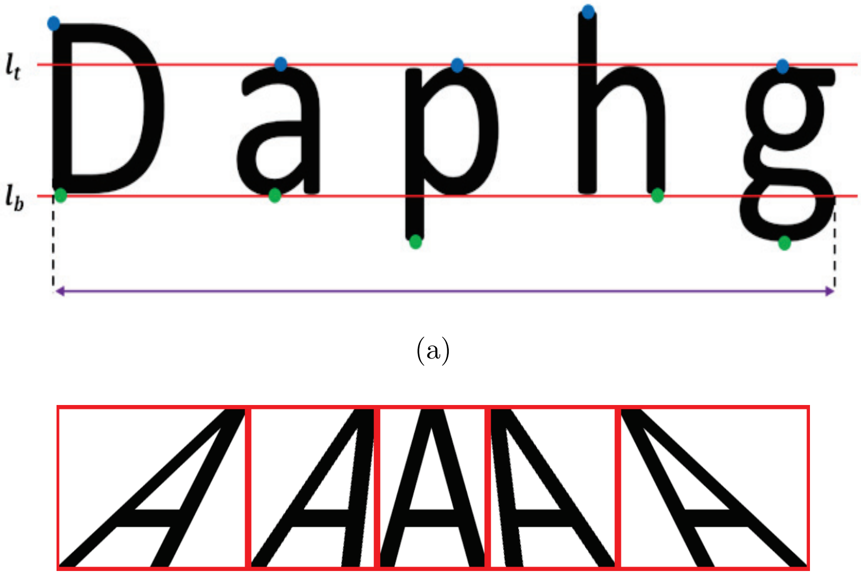


Figure 4.1: Limitations of the cost function using only glyph property. (a) Distorted input images, (b) Images minimizing the cost function using glyph property [4], (c) Rectified images by the proposed method. The proposed method exploits the alignments of characters as well as the glyph property, then yields visually pleasing results.

ops a cost function:

$$\begin{aligned} \min_{I^0, E, \tau} \; & \|I^0\|_* + \lambda_1 \|E\|_1 + f_{align}\left(\tau, \mathcal{W}\right), \\ \text{s.t.} \quad & I \circ \tau = I^0 + E, \end{aligned} \tag{4.1}$$

where $\|I^0\|_* + \lambda_1 \|E\|_1$ is a term reflecting the glyph property of rectified image, then the proposed method uses the function of low-rank transform [4]. To be precise, this term reflects the rank of texture by decomposing a transformed result $(I \circ \tau)$ of an input image (I) by transform parameters (τ), with a low-rank matrix (I^0) and a sparse error matrix (E). However, since the low-rank term does not consider the alignment of characters, the optimization of this term sometimes yields mis-aligned results. For instance, in Fig. 4.1, the low-rank cost values of images on the bottom row are 46.3, 46.1, and 47.9, respectively. For alleviating this problem, the proposed method also exploits the alignments of characters by introducing the alignment term $f_{align}\left(\tau\right)$ and alignment parameters $\mathcal{W}$.



(a)

(b)

Figure 4.2: Illustration of the character alignment properties. (a) Vertical alignment property : top (blue) and bottom (green) points of characters are on two lines. (b) Horizontal alignment property : the width of the well-rectified character has minimal value. The proposed method encodes this properties into the cost function.

4.1.2 Character alignment properties and alignment terms

Alignments of characters represent following two properties that are vertical and horizontal alignment properties. As shown in Fig. 4.2, vertical alignment means that characters in the undistorted text are aligned to the horizontal-straight lines, specifically the most top and bottom points of characters are on one of two lines. The horizontal alignment means that the character widths of undistorted text have minimal values.

In order to encode this alignment properties into the cost function, the proposed method first introduces alignment parameters $\mathcal{W} = \{(\omega_i, \mu_i)\}_{i=1}^{N}$, which are binary variables. Specifically, ω_i represents whether the top point of the i-th character is aligned to the top line l_t, and μ_i denotes whether the bottom point is aligned to the bottom line l_b. For instance, the top point of the 'D' is not on the l_t and the bottom point is on l_b as shown in Fig. 4.2, and (ω_i, μ_i) corresponding to the character in 'D' are $(0, 1)$.

For using the above stated properties, the proposed method needs to segment the characters and estimate the alignment parameters though not complete. In this section, the proposed method assumes that the results of character segment and alignment parameters are obtained, and how to decide the segmentation and alignment parameters of characters will be explained later.

Then, based on two properties of character alignments, the alignment term $f_{align}\left(\tau, \mathcal{W}\right)$ is designed as

$$f_{align}\left(\tau, \mathcal{W}\right) = \lambda_2 f_{vert}\left(\tau, \mathcal{W}\right) + \lambda_3 f_{hori}\left(\tau\right), \tag{4.2}$$

where $f_{vert}\left(\tau, \mathcal{W}\right)$ reflects the vertical alignment of characters, and $f_{hori}\left(\tau\right)$ reflects the horizontal alignment of characters.

First, the proposed method denotes a set of pixels that corresponds to the i-th character as $C_i(\tau)$, when the text image is transformed by τ. Each character segment C_i is considered a character and its top and bottom vertical positions after the transformation τ are denoted as $T_i(\tau)$ and $B_i(\tau)$, respectively:

$$\begin{aligned} T_i(\tau) &= \max_{j \in C_i(\tau)} [p_j * \tau]_y, & (4.3) \\ B_i(\tau) &= \min_{j \in C_i(\tau)} [p_j * \tau]_y, & (4.4) \end{aligned}$$

where j is a pixel index of i-th character, $p * \tau$ is the transformation of a point p by

τ, and $[p]_y$ means the vertical position of p. By using the alignment parameters and above two equations, the proposed method represents the vertical alignment term as

$$f_{vert}\left(\tau,\mathcal{W}\right)=\sum_{i=1}^{N}\omega_i\left(T_i(\tau)-y_t\right)\right)^2+\sum_{i=1}^{N}\mu_i\left(B_i(\tau)-y_b\right)^2, \tag{4.5}$$

where N is the number of character segments and y_t and y_b are the vertical positions of top and bottom lines in Fig. 4.2. As the top and bottom points of characters are on two lines, vertical alignment term has small value.

Then, the proposed method denotes width of i-th character after the transformation τ as $W_i(\tau)$:

$$W_i(\tau)=\max_{j\in C_i(\tau)}[p_j*\tau]_x-\min_{j\in C_i(\tau)}[p_j*\tau]_x, \tag{4.6}$$

where $[p]_x$ means the horizontal position of p. By using the above equation, the proposed method represents the horizontal alignment term as

$$f_{hori}\left(\tau\right)=\sum_{i=1}^{N}W_i(\tau). \tag{4.7}$$

As the sum of the character widths has small value, horizontal alignment term has small value.

As can be seen, the proposed cost function is non-convex and many variables are involved in the function. Therefore, its optimization is a difficult problem and the proposed method discusses its optimization method in the next section. Also, the overall algorithm that includes not only cost function optimization (rectification) but also character segmentation and alignment parameter estimation is mentioned in the next section.

```
Algorithm 1: Overall iterative algorithm
  Data: Input image $I$, initial transformation parameters $\tau$
  Result: Solutions ${I^0}^*$, $E^*$, $\tau^*$
  while do
      Step 1 : Determine the number of characters $N$ and character
        segmentation results $\{C_i(\tau)\}_{i=1}^N$
      Step 2 : Estimate the alignment parameters $\mathcal{W}$
      Step 3 : Estimate the rectification variables ($I^0$, $E$, $\tau$) by minimizing the
        cost function (4.1)
  end
```

4.2 Overall algorithm

For the alignment term in (4.1), the proposed method needs to segment the text into individual characters and determine their alignment parameters. The character segmentation is a difficult problem in the case of distorted images. Since better segmentation needs better rectification and vice versa, the proposed method solves this problem by performing the character segmentation (and alignment parameter estimation) and rectification iteratively. As shown in Fig. 4.3, the proposed method confirms that scene text rectification and character segmentation are performed as performing the proposed iterative scheme more and more. For the iterative scheme, first the proposed method performs the character segmentation in the rectified image (by the current estimated τ) using a projection profile method. After character segmentation, the proposed method estimates the alignment parameters by using a least squares line fitting with RANSAC outlier removal. Then, the proposed method performs rectification by minimizing the cost function in (4.1). Since the cost function is non-convex function and many variables are involved in the function, it is not straightforward to minimize the cost function. To solve this problem, the proposed method adds the auxiliary variables and find solution by solving the linearized prob-

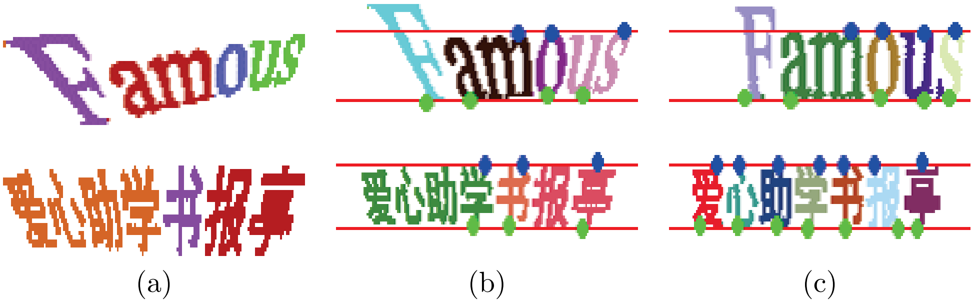


Figure 4.3: The proposed iterative rectification and segmentation scheme. (a) Input distorted images and their character segments, (b) Rectified images and their character segments after first iteration. (c) Rectified images and their character segments after second iteration. Top (blue) points and Bottom (green) points are aligned to two horizontal (red) lines, and characters are better rectified and segmented as performing iterative scheme.

lem iteratively. The above mentioned three processes are performed iteratively. The whole optimization process is summarized in Algorithm 1.

4.2.1 Initialization

For the initialization of transformation parameters τ, the proposed method estimates rough skew and shearing angles. First, for the skew estimation, the proposed method computes the projection profiles and considers the angle yielding the most compact profile as the skew angle θ_r. After correcting the skew with the estimated angle, the proposed method estimates the shearing angle θ_t by maximizing the length sum of zero-runs in horizontal profiles (i.e., intervals between CCs) in a search range of $(-\pi/3, \pi/3)$. Then, the initial homography matrix is given by

$$\begin{pmatrix} 1 & -\tan\theta_t & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\theta_r & -\sin\theta_r & 0 \\ \sin\theta_r & \cos\theta_r & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{4.8}$$

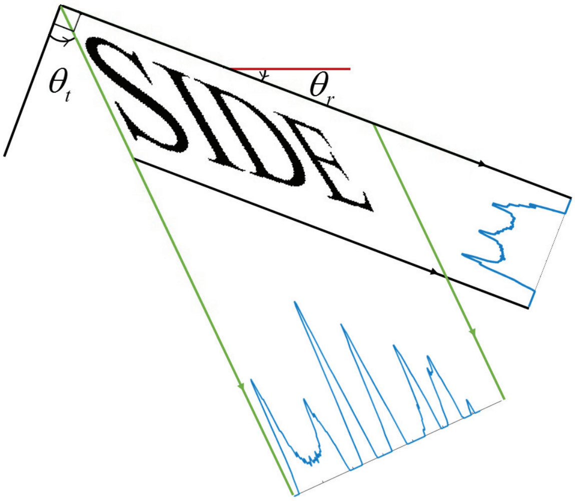
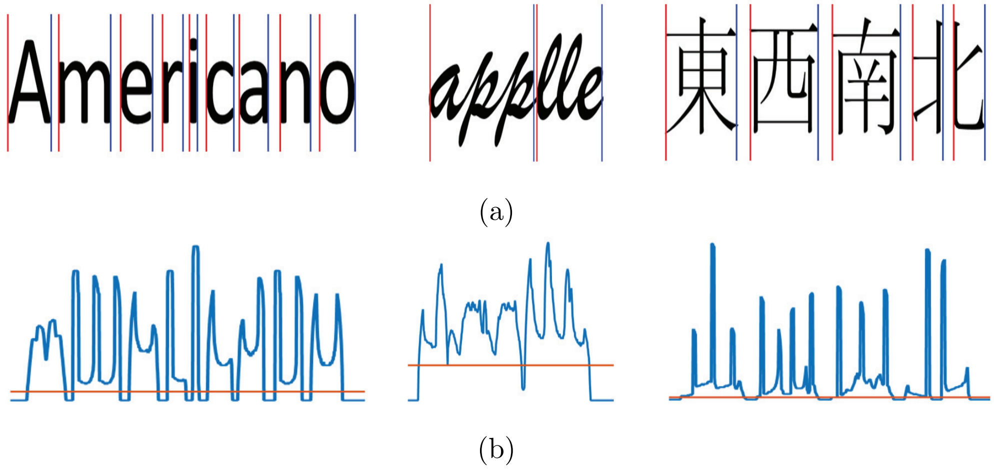


Figure 4.4: Estimation of the skew and shearing angles using the projection profile analysis.

and a transformation parameter vector τ is initialized by this homography. This step is illustrated in Fig. 4.4.

4.2.2 Character segmentation

In order to encode the alignments of characters into the proposed cost function, the proposed method separates scene text into individual characters. First, the proposed method extracts connected components (CCs), and estimates the mean stroke width of scene text. Intuitively, the stroke width represents how far the pen moves to write a given CC. For this representation, the proposed method estimates the stroke widths of all CCs [58], and calculates the mean stroke width. Second, the proposed method computes a horizontal projection profile and separates the characters with the projection value of more than mean stroke width value. As shown in Fig. 4.5, all scene text including English and Chinese characters are separated by this projection



(a)

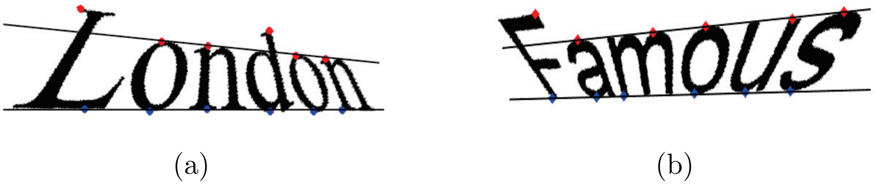
(b)

Figure 4.5: Results of character segmentation. (a) Character segments. (b) (Blue) graph and horizontal (red) lines mean projection profiles and mean stroke width values, respectively.

profile method. Also, Chinese characters are sometimes over-segmented, however the proposed method confirms that segmentation is correctly performed for the purpose, because the over-segmented characters are satisfied with the character alignment properties. Also, the characters of the cursive script are under-segmented, since all the characters are a little tilt. Then, the proposed method determines the horizontal positions $C_i(\tau)$ using the result of character segmentation.

4.2.3 Estimation of the alignment parameters

The alignment parameters determine whether the corresponding characters are aligned to top and bottom lines. However, since the current estimated τ is still not perfectly optimized, the characters in the rectified image (by the current estimated τ) are not aligned to two horizontal lines l_t and l_b. In the case of the projective transformation, when the three points on the same line are transformed, they still lie on the same line in the transformed domain (line to line mapping). Therefore, although horizontal



(a) (b)

Figure 4.6: Results of line fitting on computed top and bottom points with RANSAC.

lines l_t and l_b are not horizontal, however they are still straight in the rectified image (by the current estimated τ). These two (maybe not horizontal but slant) lines are obtained by performing least squares line fitting on computed top and bottom points with RANSAC outlier removal, and alignment parameters are also estimated. For instance, as shown in Fig. 4.6-(a), only four top points (corresponding to 'o' and 'n') are aligned. Hence, alignment parameters ω_i corresponding to the top line ('o' and 'n') are 1, and the others ('L' and 'd') are 0. Also, since all bottom points are on bottom line, all alignment parameter μ_i are 1.

4.2.4 Cost function optimization for rectification

The proposed method estimates the variables (I^0, E, τ) by minimizing the proposed cost function. For this, the proposed method develops an optimization method for (4.1), which is non-convex and thus difficult to optimize. Therefore, the proposed method first transforms the cost function into a convex function by introducing auxiliary variables $\Gamma = \{(\alpha_i, \beta_i, \gamma_i)\}_{i=1}^{N}$:

$$\begin{aligned} \min_{I^0,E,\tau,\Gamma} \|I^0\|_* + \lambda_1 \|E\|_1 + \lambda_2 \sum_{i=1}^{N} \omega_i(\alpha_i - y_t)^2 + \lambda_2 \sum_{i=1}^{N} \mu_i(\beta_i - y_b)^2 + \lambda_3 \sum_{i=1}^{N} \gamma_i, & \\ \text{s.t.} \quad I \circ \tau = I^0 + E, & \\ T_i(\tau) = \alpha_i, & \\ B_i(\tau) = \beta_i, & \\ W_i(\tau) = \gamma_i. & \end{aligned} \tag{4.9}$$

Then, the proposed method linearizes the constraints around the current estimated value τ and solves the linearized problem [4]. The linearized problem becomes

$$\begin{aligned} \min_{I^0,E,\Delta\tau,\Gamma} \|I^0\|_* + \lambda_1 \|E\|_1 + \lambda_2 \sum_{i=1}^{N} \omega_i(\alpha_i - y_1)^2 + \lambda_2 \sum_{i=1}^{N} \mu_i(\beta_i - y_b)^2 + \lambda_3 \sum_{i=1}^{N} \gamma_i, & \\ \text{s.t.} \quad \nabla I \Delta\tau + I \circ \tau = I^0 + E, & \\ \nabla T_i \Delta\tau + \max_{j \in C_i(\tau)} [p_j * \tau]_y = \alpha_i, & \\ \nabla B_i \Delta\tau + \min_{j \in C_i(\tau)} [p_j * \tau]_y = \beta_i, & \\ \nabla W_i \Delta\tau + \max_{j \in C_i(\tau)} [p_j * \tau]_x - \min_{j \in C_i(\tau)} [p_j * \tau]_x = \gamma_i, & \end{aligned} \tag{4.10}$$

where $\nabla I = \frac{\partial(I \circ \tau)}{\partial \tau}$, $\nabla T_i = \frac{\partial(T_i(\tau))}{\partial \tau}$, $\nabla B_i = \frac{\partial(B_i(\tau))}{\partial \tau}$, and $\nabla W_i = \frac{\partial(W_i(\tau))}{\partial \tau}$.

To solve the above linearized convex problem, the proposed method uses the Augmented Lagrange Multiplier (ALM) method [59]. The proposed method can find the global optimum of the linearized problem by using the ALM method as in [4]. However, since (4.10) is a local approximation of the original non-convex problem,

Algorithm 2: Solving the problem (4.9)

```
Data: Input image $I$, initial transformation parameters $\tau$, the number of
      characters $N$, character segments $\{C_i\}_{i=1}^{N}$, alignment parameters
      $\mathcal{W} = \{(\omega_i, \mu_i)\}_{i=1}^{N}$.
Result: Solution ${I^0}^*$, $E^*$, $\tau^*$
while not converged do
    Step 1 : Compute Jacobians $\nabla I$, $\nabla T_i$, $\nabla B_i$, and $\nabla W_i$
    Step 2 : Solve the linearized problem in (4.10)
    Step 3 : Update the transformation: $\tau \leftarrow \tau + \Delta\tau$
end
```

the proposed method finds the solution of the original problem (4.9) by solving the linearized problem iteratively. This iterative solver is summarized in Algorithm 2.

Review of the ALM method

The general constrained convex problem is like that:

$$\min_{X} f(X), \quad \text{s.t.} \quad R(X) = 0, \tag{4.11}$$

where f is convex function of variables X, and $R(X) = 0$ is a linear constraint. The ALM method converts constrained convex problem into unconstrained convex problem as follow:

$$\begin{gathered} L_\eta(X, Y) = f(X) + < Y, R(X) > + \frac{\eta}{2} {\|R(x)\|_2}^2, \\ \min_{X} L_\eta(X, Y), \end{gathered} \tag{4.12}$$

where Y is a Lagrange multiplier and $\eta > 0$ is the penalty imposed for feasibility. Two above problems in (4.11), (4.12) have same optimal solution, when Lagrange multiplier Y has an appropriate value and penalty η is sufficiently large. The ALM method solves the constrained convex problem in (4.11) by optimizing the augmented convex problem in (4.12). The ALM method optimizes the augmented Lagrangian function and estimates an appropriate Lagrange multiplier as following iterative step:

$$\begin{aligned} X_{k+1} &= \arg\min_X L_{\mu_k}(X, Y_k), \\ Y_{k+1} &= Y_K + \eta_k(R(X_k)), \\ \eta_{k+1} &= \rho \cdot \eta_k, \end{aligned} \tag{4.13}$$

where $\eta > 1$ is a parameter increasing penalty η.

Solving linearized problem

In the proposed optimization, the input variable X, object function $f(\cdot)$ and linear constraints $R(\cdot)$ become

$$\begin{aligned}&X = (I^0, E, \Delta\tau, \Gamma),\\ &f(X) = \|I^0\|_* + \lambda_1\|E\|_1 + \lambda_2\sum_{i=1}^{N}\omega_i(\alpha_i - y_t)^2 + \lambda_2\sum_{i=1}^{N}\mu_i(\beta_i - y_b)^2 + \lambda_3\sum_{i=1}^{N}\gamma_i,\end{aligned}$$

$$R(X) = \left(\begin{array}{c} \nabla I\Delta\tau + I\circ\tau - I^0 + E \\ \\ \left[\left(\nabla T_i\Delta\tau + \max_{j\in C_i(\tau)}[p_j * \tau]_y - \alpha_i\right)_{i=1}^{N}\right]^\top \\ \\ \left[\left(\nabla B_i\Delta\tau + \min_{j\in C_i(\tau)}[p_j * \tau]_y - \beta_i\right)_{i=1}^{N}\right]^\top \\ \\ \left[\left(\nabla W_i\Delta\tau + \max_{j\in C_i(\tau)}[p_j * \tau]_x - \min_{j\in C_i(\tau)}[p_j * \tau]_x - \gamma_i\right)_{i=1}^{N}\right]^\top \end{array}\right), \tag{4.14}$$

where $\left[(x_i)_{i=1}^{N}\right]^\top$ mean the vectors of $[x_1, x_2, ..., x_N]^\top$.

The proposed method estimates the optimal solutions $(I^0, E, \Delta\tau, \Gamma)$ by performing iterative step of ALM method in (4.13). The updating Lagrange multiplier and penalty is simple, the proposed method mentions only the first step updating the variable X of the iterative step in (4.13), in detail.

Since many variables are involved in the augmented Lagrangian function, it is unstraightforward process to minimize the augmented Lagrangian function. To solve this problem, the proposed method develops an alternating optimization method [4] as follow:

$$
\begin{aligned}
I^0_{k+1} &= \arg\min_{I^0} L_{\eta_k}\left(I^0, E_k, \Delta\tau_k, \Gamma_k, Y_k\right), \\
E_{k+1} &= \arg\min_{E} L_{\eta_k}\left(I^0_{k+1}, E, \Delta\tau_k, \Gamma_k, Y_k\right), \\
\Gamma_{k+1} &= \arg\min_{\Gamma} L_{\eta_k}\left(I^0_{k+1}, E_{k+1}, \Delta\tau_k, \Gamma, Y_k\right), \\
\Delta\tau_{k+1} &= \arg\min_{\Delta\tau} L_{\eta_k}\left(I^0_{k+1}, E_{k+1}, \Delta\tau, \Gamma_{k+1}, Y_k\right).
\end{aligned}
\tag{4.15}
$$

The proposed method solves the above problems by minimizing the low-rank and alignment terms alternatively, and they have closed-form solutions. The solution I^0, E and $\Delta\tau$ are mentioned in [4], and the solutions for the auxiliary additional variables become:

$$
\begin{aligned}
\alpha_{k+1} &\leftarrow \begin{cases} \frac{1}{2\lambda_2+\eta_k}(2\lambda_2 y_t + Y_k^{\alpha}\eta_k \nabla T \Delta\tau_k + \max_{j\in C(\tau)}[p_j * \tau]_y), & \omega_i = 1 \\ \max_{j\in C(\tau)}[p_j * \tau]_y, & \text{otherwise} \end{cases} \\
\beta_{k+1} &\leftarrow \begin{cases} \frac{1}{2\lambda_2+\eta_k}(2\lambda_2 y_b + Y_k^{\beta}\eta_k \nabla T \Delta\tau_k + \min_{j\in C(\tau)}[p_j * \tau]_y), & \mu_i = 1 \\ \min_{j\in C(\tau)}[p_j * \tau]_y, & \text{otherwise} \end{cases} \\
\gamma_{k+1} &\leftarrow \frac{(-\lambda_3 + Y_k^{\gamma})}{\eta_k} + \nabla W \Delta\tau_k + \max_{j\in C(\tau)}[p_j * \tau]_x - \min_{j\in C(\tau)}[p_j * \tau]_x,
\end{aligned}
\tag{4.16}
$$

where k is an iteration index, Y_α, Y_β and Y_γ are the Lagrangian multiplier corresponding to linear constraints related to α, β, and γ.

4.3 Experimental results

This dissertation evaluated the proposed scene text rectification method on the scene text dataset containing real scene text images and synthetic text images [43]. The

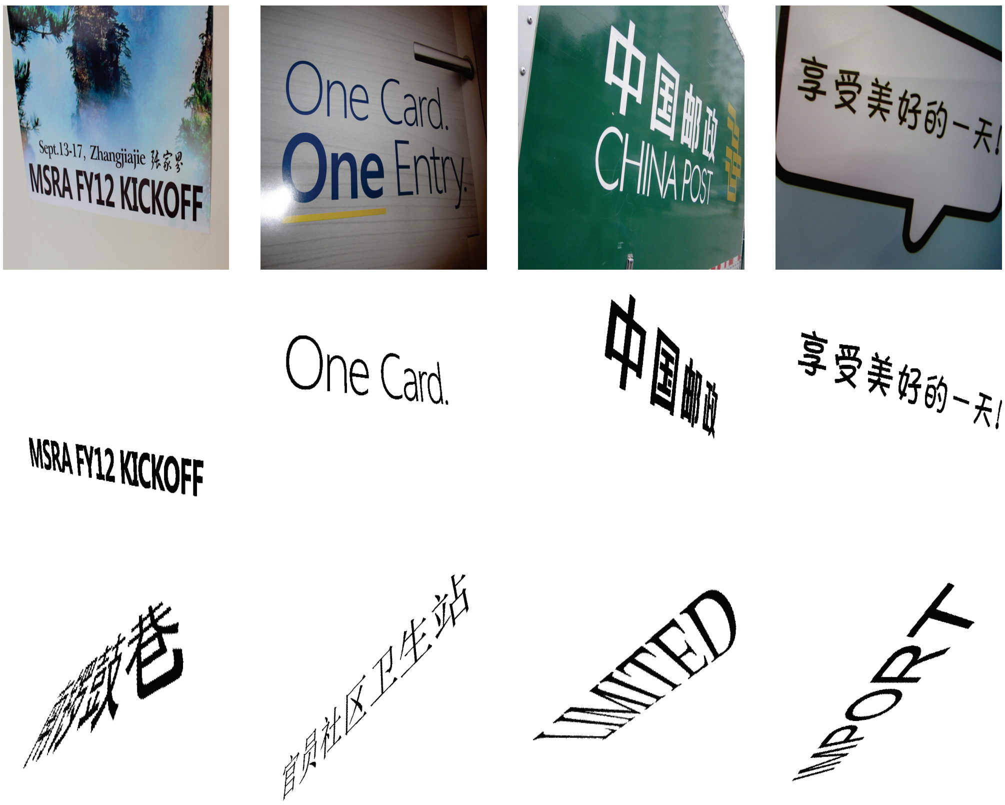


Figure 4.7: Images of dataset. The first row: real scenes, the second row: detection and binarization results of first row, the third row: synthetic text images.

real scene text images were from MSRA-TD 500 dataset [60] and synthetic text images were obtained by applying homography transformation to ICDAR 2013 Robust Reading Competition dataset [61]. They include English and Chinese characters, as shown in Fig. 4.7.

In the experiments, the proposed method resized inputs so that their pixel-resolutions became 3500 pixels. The proposed method set the weight λ_2, λ_3 in 4.1 so that it is proportional to $1/N$, where N is the number of character segments. Also, for the image height h, the proposed method set $y_t = 0$ and $y_b = h$. The proposed method implemented with C++ and its implementation takes 2-6 s for the rectification of an image (3500-pixel resolution) on Intel(R) i5(TM) CPU(3.40 GHz).

For the objective evaluations of the rectification, the conventional method [4] is compared with the proposed method. The performance is evaluated in terms of OCR accuracy. To be precise, the accuracy of OCR is defined as

$$\text{accuracy}(R, G) = 1 - \frac{L(R, G)}{\max(\#R, \#G)}, \tag{4.17}$$

where R is a recognition result string, G is the ground truth string, $\#(\cdot)$ is the number of characters in the string, and $L(x, y)$ means the Levenshtein distance between two strings [57]. The distance is defined as the minimum number of character edits (insertion, deletion, and substitution) to transform one string to the other. For the OCR, the proposed method used the google tesseract OCR engine [7].

Experimental results for OCR accuracy are summarized in Table. 4.1 and Table. 4.2. As shown, the proposed method shows higher accuracy for both real and synthetic text images. Since real text images consist of relatively less distorted images, the difference between the conventional method [4] and the proposed method is relatively small compared with synthetic images.

Some experimental results are shown in Fig. 4.8, 4.9, 4.10 and 4.11. The conventional method [4] has difficulties in the severe perspective distortions, but the proposed method corrects severe perspective distortions as well as the little perspective distortions well. Also, since the proposed method adds the alignment term, characters are more aligned in the proposed method compared with the conventional method in [4].

Table 4.1: OCR accuracy of the proposed and conventional method [4] on real scene text images.

OCR accuracy	English	Chinese
Input image	0.6128	0.4566
TILT [4]	0.9178	0.6326
Proposed	**0.9421**	**0.7183**

Table 4.2: OCR accuracy of the proposed and conventional method [4] on synthetic text images.

OCR accuracy	English	Chinese
Input image	0.1475	0.1178
TILT [4]	0.7974	0.4005
Proposed	**0.9023**	**0.6531**

4.4 Summary

In this chapter, this dissertation has proposed a new scene text rectification algorithm. In the proposed method, two properties of rectified text images are encoded into the cost function and the proposed method obtained optimal transformation parameters by minimizing the cost function. Since the proposed method considered the alignments of characters, the proposed algorithm yielded improved rectification performance for a range of cases. Also, in order to encode the alignments of characters into the proposed function, the proposed method separated the scene text into individual characters. Overall algorithm was designed as performing the character segmentation (and alignment parameter estimation) and rectification iteratively. Experimental results on natural and synthetic images showed that the OCR accuracy of the proposed algorithm is higher than the conventional methods using only glyph properties.

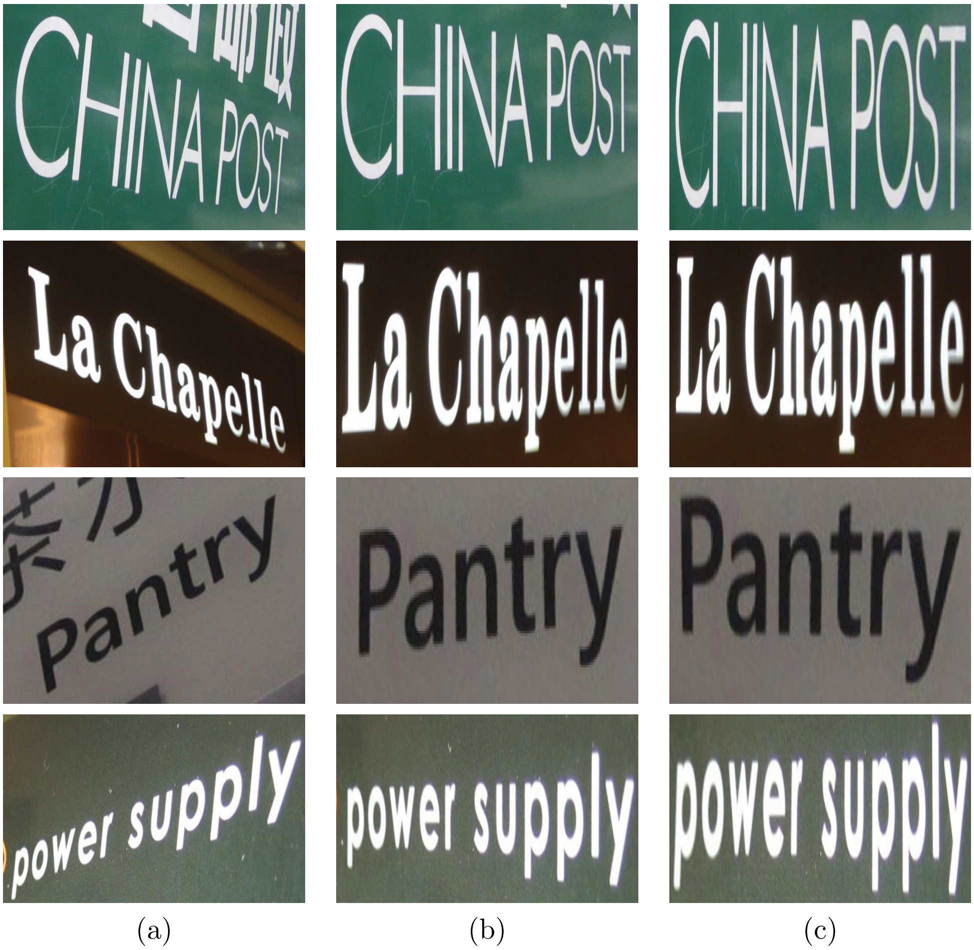
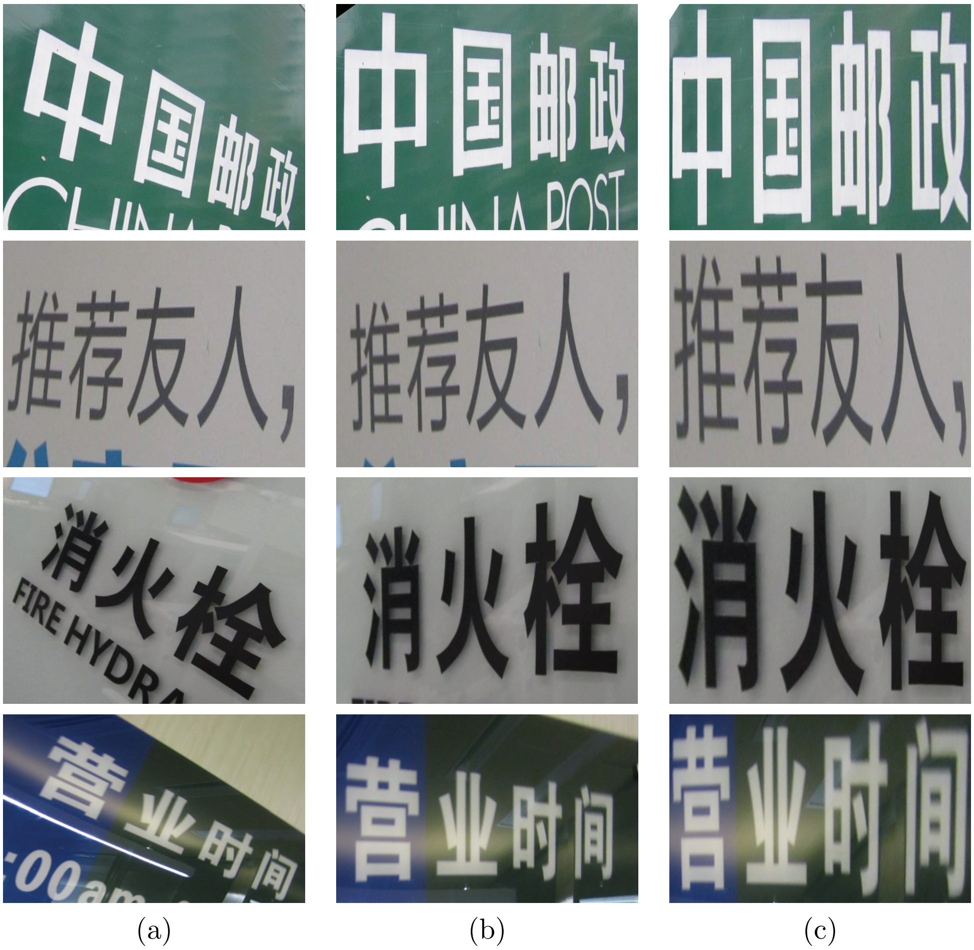
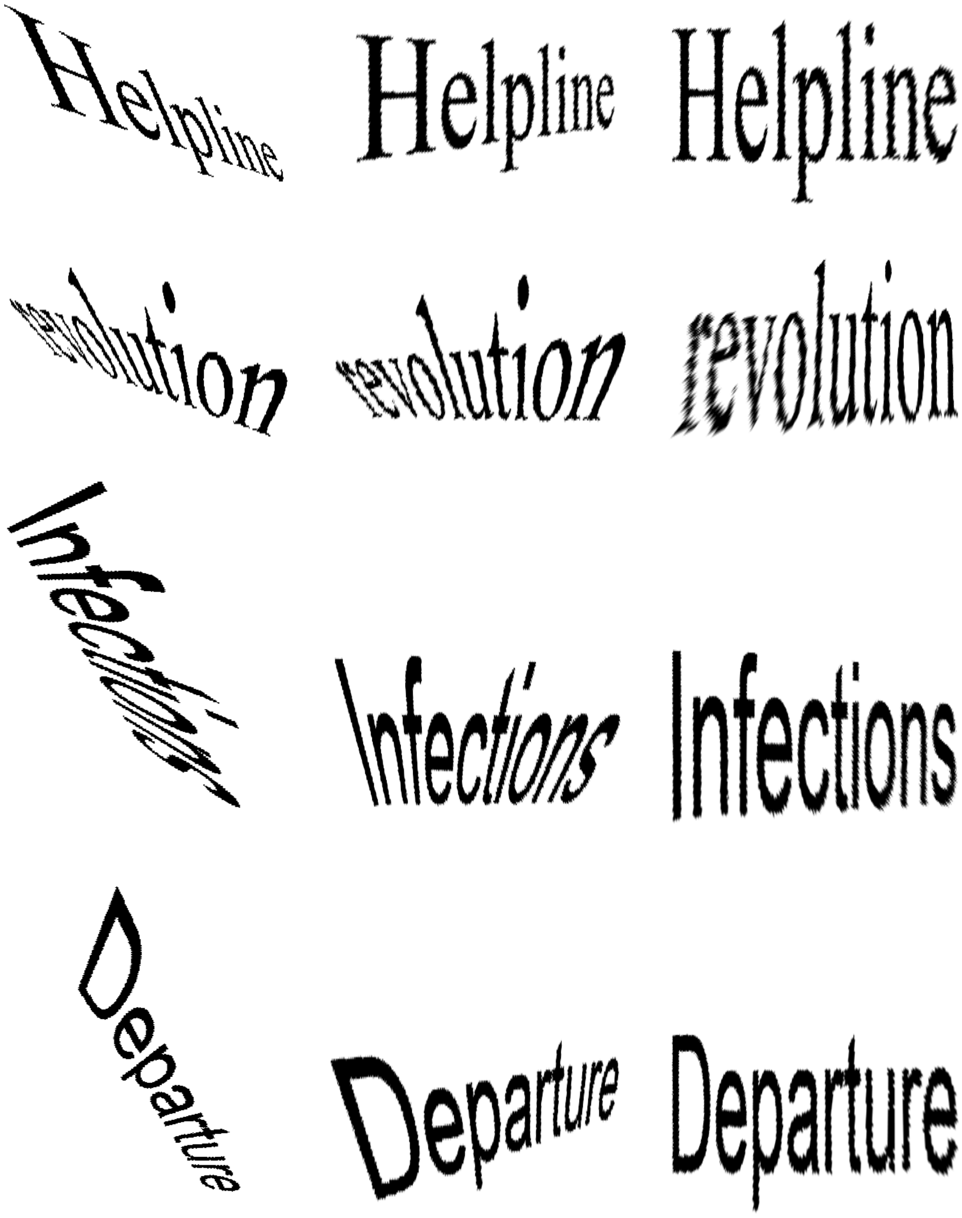


Figure 4.8: Rectification results on English real scene text images. (a) input distorted image, (b) the conventional method [4], (c) the proposed method.



(a) (b) (c)

Figure 4.9: Rectification results on Chinese real scene text images. (a) input distorted image, (b) the conventional method [4], (c) the proposed method.



(a) (b) (c)

Figure 4.10: Rectification results on English synthetic text images. (a) input distorted image, (b) the conventional method [4], (c) the proposed method.

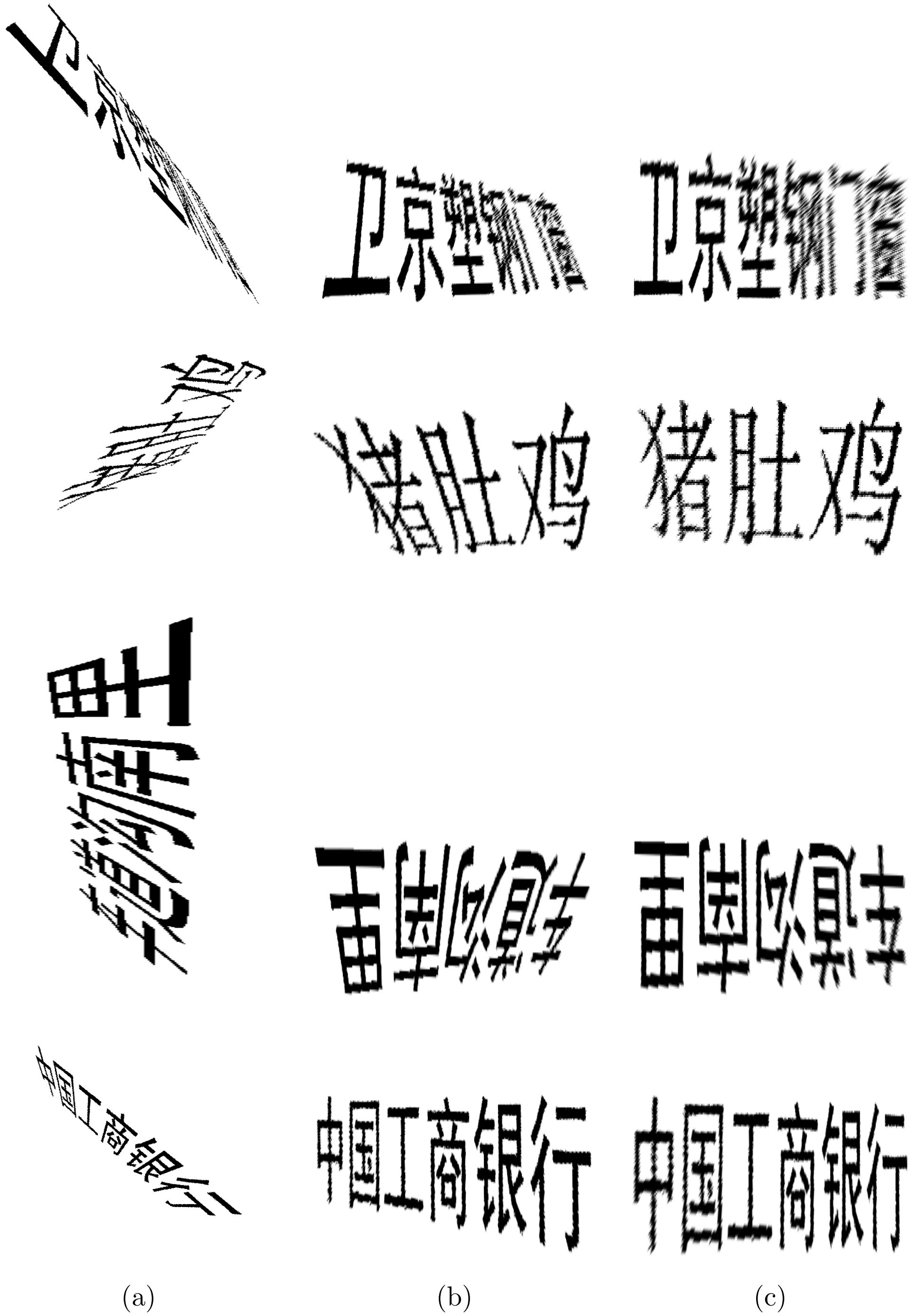


Figure 4.11: Rectification results on Chinese synthetic text images. (a) input distorted image, (b) the conventional method [4], (c) the proposed method.

Chapter 5

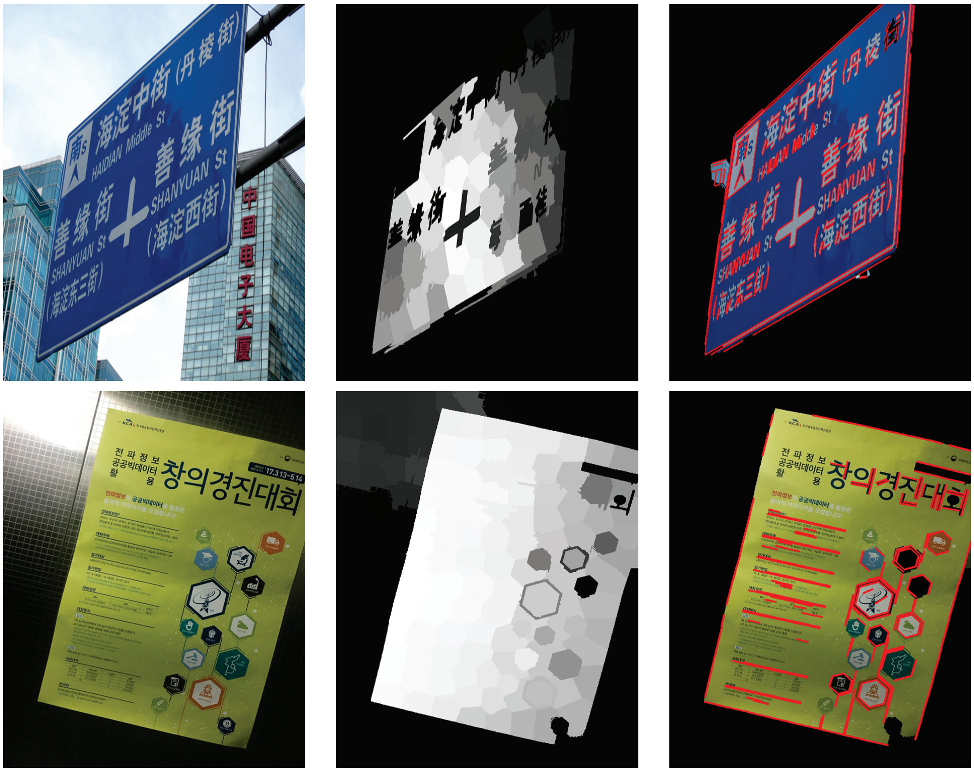
Curved surface dewarping in real scene

5.1 Proposed curved surface dewarping method

The proposed cost function in document dewarping can be extended to curved surface dewarping in real scene. In order to adapt to real scene images, a pre-processing step is introduced, in this section.

5.1.1 Pre-processing

In order to rectify curved surface in real scene, the proposed method first detects the salient planar objects. There are many salient object detection methods, the proposed method detects the salient planar objects using [5]. This saliency detection methods ranks the similarity of the image elements with foreground cues or back ground cures via graph-based manifold ranking, then defined the saliency of the image elements as their relevances to the given seeds or queries. Then, the proposed



(a) (b) (c)

Figure 5.1: Results of the saliency detection [5]. (a) Input images, (b) Saliency maps, (c) Results of feature extraction on salient objects.

method extracts color pixels and line segments on the only salient regions. This results is as shown in Fig. 5.1.

5.2 Experimental results

First, in order to evaluate the effectiveness of pre-processing, this dissertation compared three images: results of [4] in the whole image, results of [4] in the salient object, the results of proposed method. These experimental results are shown in Fig. 5.3, 5.4 and 5.5. By exploiting the saliency detection, object rectification in real

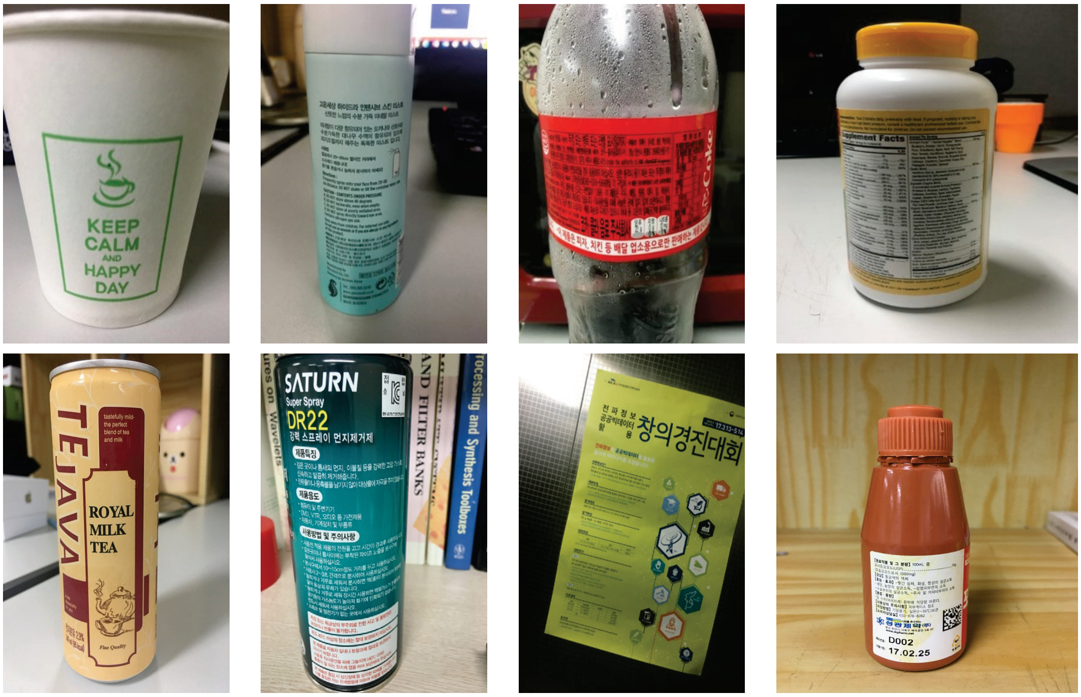


Figure 5.2: Images of the curved surface in a real scene dataset collected by this dissertation.

scene is more aligned, as shown in Fig. 5.3, 5.4 and 5.5-(b) and (c).

Also, in order to consist of curved surface images in real scene, the proposed method collected 74 curved surface images as shown in the Fig. 5.2. For the evaluation, the proposed method computed the geometric quantities of rectangles in rectified results mentioned in scene text rectification. Experimental results are summarized in Table. 5.1. The existing methods [1, 6] are compared with the proposed method. As shown, the proposed method shows improved geometric rectification performance in terms of all measures.

Some experimental results are shown in Fig. 5.6, Fig. 5.7, and Fig. 5.8. Unlike the proposed method, the conventional text-line based method has difficulties by a

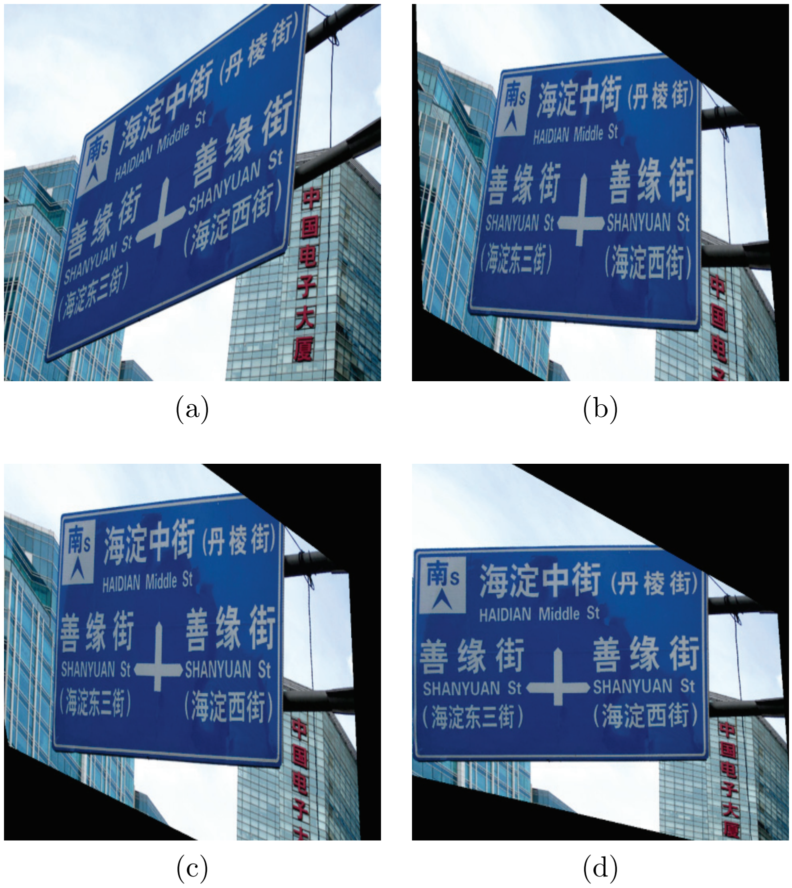
Table 5.1: Geometric measures of the proposed and conventional methods [1, 6] on our dataset

	Kim [1]	Zhang [6]	Proposed
Orthogonality	11.3624	8.6529	**4.5710**
Diagonal ratio	0.0414	0.0317	**0.0124**
Vertical ratio	0.0916	0.0655	**0.0498**
Horizontal ratio	0.0861	0.0655	**0.0480**

few text-lines and their false positives. Also, the low-rank based method works well in well-structured images such as barcode, however yield distortions in other images. However, since the proposed method exploits well aligned line segments including boundary lines of circular objects, the proposed method works well.

5.3 Summary

In this chapter, this dissertation has proposed a dewarping algorithm of curved surface in real scene by extending the document dewarping algorithm in 3. Since the curved surface in real scene contains a lots of line segments (i.e., boundary lines of circular objects or barcode lines), the proposed algorithm yielded improved rectification performance for a range of cases. Experimental results on the tested dataset showed that the geometric measures of the proposed algorithm is higher than the conventional methods.



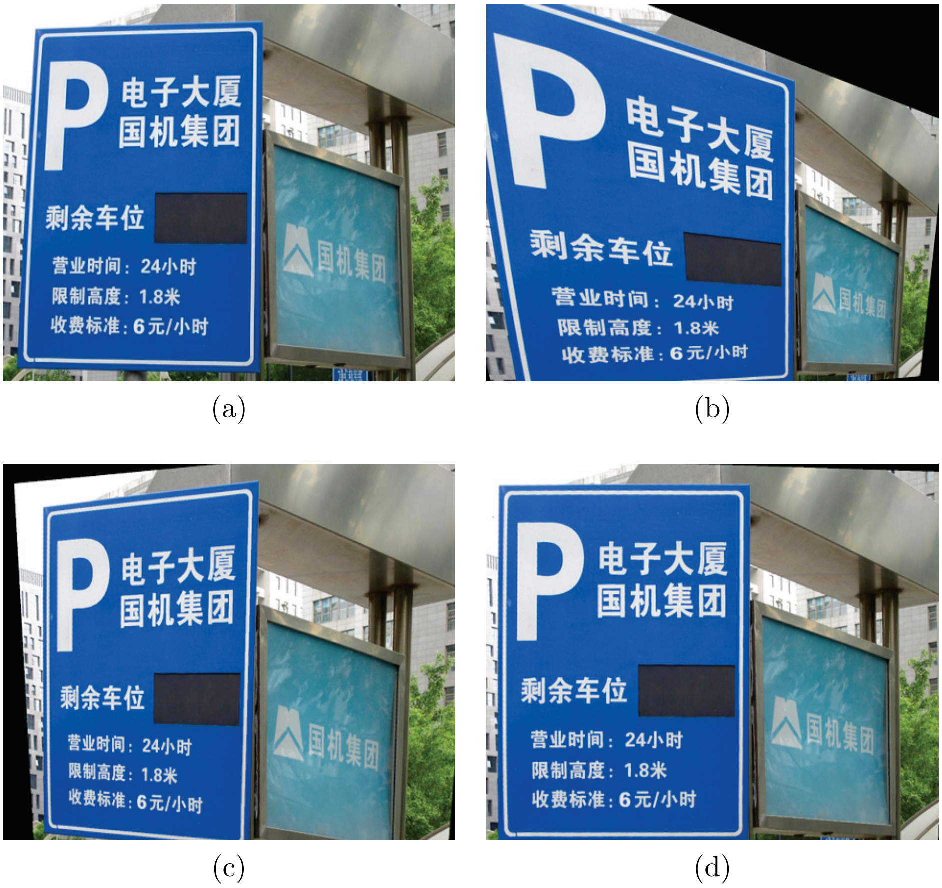
(a)

(b)

(c)

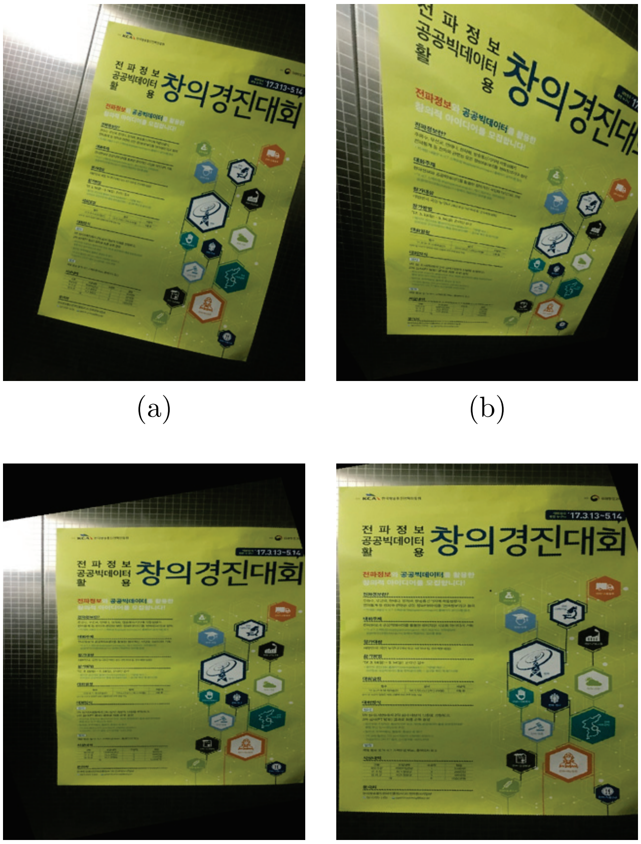
(d)

Figure 5.3: Results of the three methods. (a) Input image, (b) Result of [4], (c) Result of [4] for the salient object region, (d) Result of the proposed method.



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

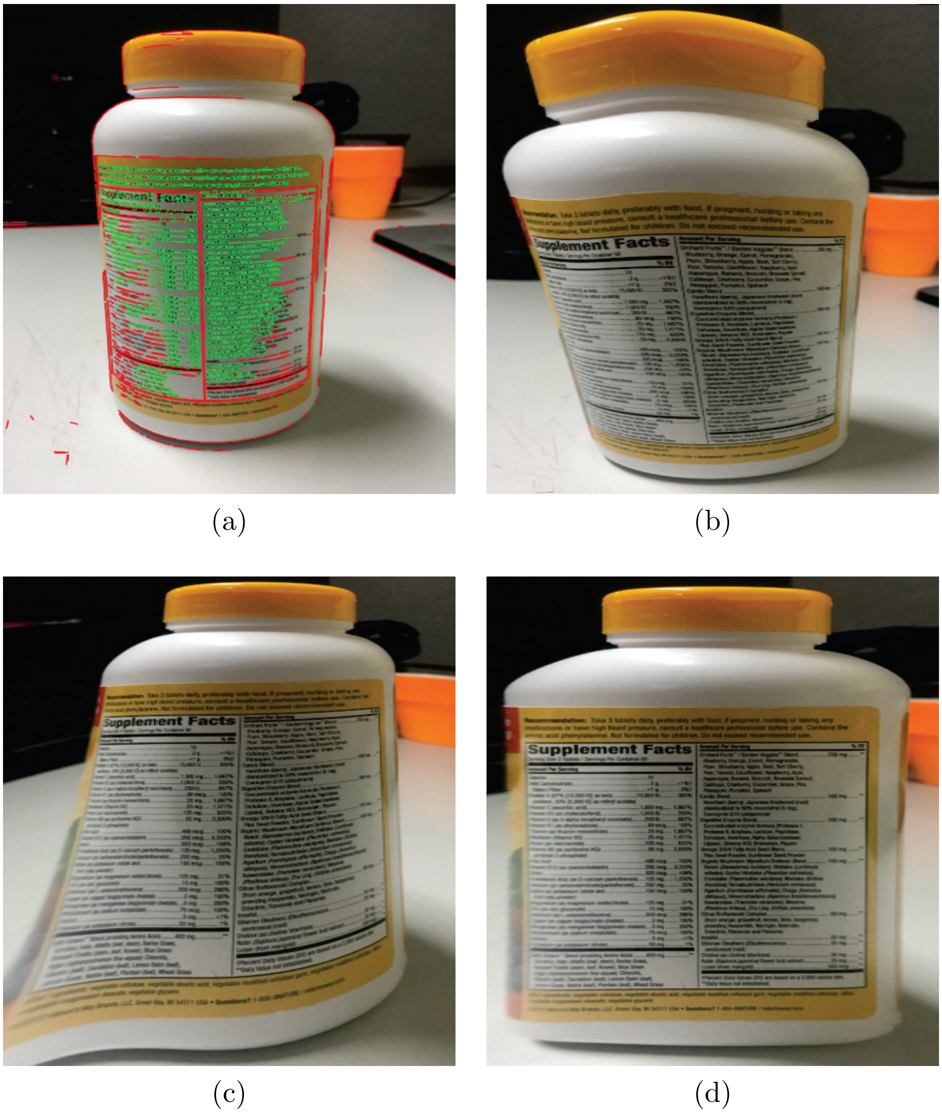
Figure 5.4: Results of the three methods. (a) Input image, (b) Result of [4], (c) Result of [4] for the salient object region, (d) Result of the proposed method.



(a) (b)

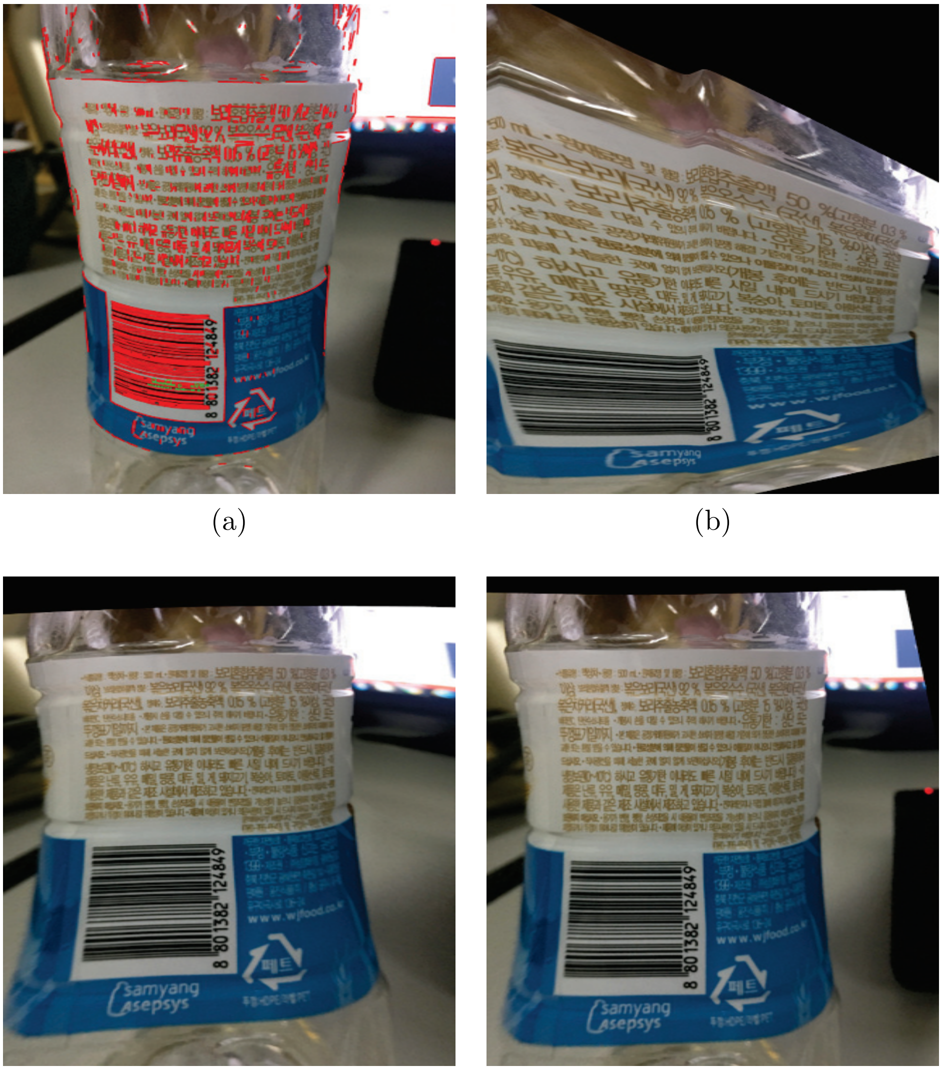
(c) (d)

Figure 5.5: Results of the three methods. (a) Input image, (b) Result of [4], (c) Result of [4] for the salient object region, (d) Result of the proposed method.



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

Figure 5.6: Results of the three methods. (a) Input image, (b) Result of [1], (c) Result of [6], (d) Result of the proposed method.



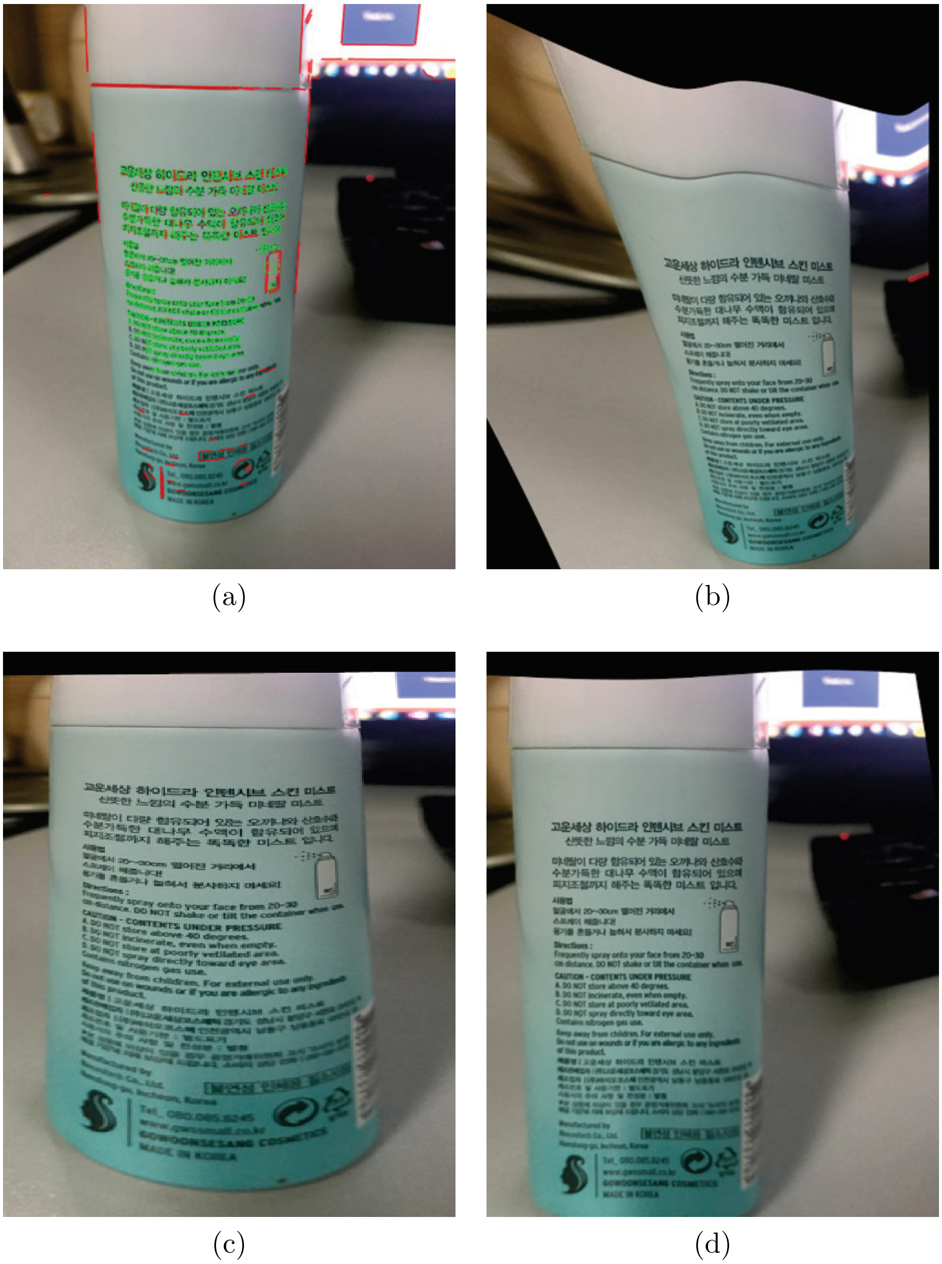
(a)

(b)

(c)

(d)

Figure 5.7: Results of the three methods. (a) Input image, (b) Result of [1], (c) Result of [6], (d) Result of the proposed method.



(a) (b)

(c) (d)

Figure 5.8: Results of the three methods. (a) Input image, (b) Result of [1], (c) Result of [6], (d) Result of the proposed method.

Chapter 6

Conclusions

In this dissertation, a new rectification method for document and scene text image based on alignment properties have been proposed. For document image dewarping, the proposed method exploited line segment properties which are valid in non-text as well as text regions, encoded this properties into a cost function and obtained optimum transformation parameters by minimizing the cost function. Since the proposed method considered the properties on non-text as well as text regions, the proposed algorithm yields improved OCR accuracy and geometric measures, especially for the images include many non-text regions. For scene text rectification, in the proposed method, two properties of rectified text images were encoded into the cost function and also obtained transformation parameters by minimizing the cost function. Since the proposed method considered the alignments of characters as well as glyph (low-rank) property, the proposed algorithm yielded improved rectification performance for a range of cases. Experimental results on natural and synthetic images showed that the OCR accuracy of the proposed algorithm is higher than the conventional low-rank based method. In addition, the proposed alignment term of line segments was extended to the curved surface dewarping in a real scene.

Bibliography

[1] B. S. Kim, H. I. Koo, and N. I. Cho, "Document dewarping via text-line based optimization," *Pattern Recognition*, vol. 48, no. 11, pp. 3600–3614, 2015.

[2] R. G. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "LSD: a line segment detector," *Image Processing On Line*, vol. 2, pp. 35–55, 2012.

[3] S. S. Bukhari, F. Shafait, and T. M. Breuel, "Dewarping of document images using coupled-snakes," in *Proceeding of International Workshop on Camera Based Document Analysis and Recognition*, 2009, pp. 34–41.

[4] Z. Zhang, X. Liang, A. Ganesh, and Y. Ma, "TILT: transform invariant low-rank textures," in *Asian Conference on Computer Vision*, 2010, pp. 314–328.

[5] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3166–3173.

[6] Z. Zhang, X. Liang, and Y. Ma, "Unwrapping low-rank textures on generalized cylindrical surfaces," in *IEEE International Conference on Computer Vision*, 2011, pp. 1347–1354.

[7] R. Smith, “An overview of the tesseract ocr engine,” in *International Conference on Document Analysis and Recognition*, vol. 2, 2007, pp. 629–633.

[8] T. A. Tran, I. S. Na, and S. H. Kim, “Page segmentation using minimum homogeneity algorithm and adaptive mathematical morphology,” *International Journal on Document Analysis and Recognition*, vol. 19, no. 3, pp. 191–209, 2016.

[9] H. I. Koo and D. H. Kim, “Scene text detection via connected component clustering and nontext filtering,” *IEEE Transactions on Image Processing*, vol. 22, no. 6, pp. 2296–2305, 2013.

[10] L. O’Gorman, “The document spectrum for page layout analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1162–1173, 1993.

[11] G. Nagy, “Twenty years of document image analysis in pami,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 38–62, 2000.

[12] H. I. Koo, “Text-line detection in camera-captured document images using the state estimation of connected components,” *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5358–5368, 2016.

[13] L.-Y. Duan, R. Ji, Z. Chen, T. Huang, and W. Gao, “Towards mobile document image retrieval for digital library,” *IEEE Transactions on Multimedia*, vol. 16, no. 2, pp. 346–359, 2014.

[14] K. Levenberg, “A method for the solution of certain non-linear problems in least squares,” *Quarterly of Applied Mathematics*, vol. 2, no. 2, pp. 164–168, 1944.

[15] D. W. Marquardt, “An algorithm for least-squares estimation of nonlinear parameters,” *Journal of the society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.

[16] Y. Lu, “Machine printed character segmentation—; an overview,” *Pattern Recognition*, vol. 28, no. 1, pp. 67–80, 1995.

[17] R. G. Casey and E. Lecolinet, “A survey of methods and strategies in character segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 7, pp. 690–706, 1996.

[18] M. S. Brown, M. Sun, R. Yang, L. Yun, and W. B. Seales, “Restoring 2d content from distorted documents,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 11, 2007.

[19] L. Zhang, Y. Zhang, and C. Tan, “An improved physically-based method for geometric restoration of distorted document images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 4, pp. 728–734, 2008.

[20] O. Samko, Y.-K. Lai, D. Marshall, and P. L. Rosin, “Virtual unrolling and information recovery from scanned scrolled historical documents,” *Pattern Recognition*, vol. 47, no. 1, pp. 248–259, 2014.

[21] G. Meng, Y. Wang, S. Qu, S. Xiang, and C. Pan, “Active flattening of curved document images via two structured beams,” in *Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3890–3897.

[22] Y.-C. Tsoi and M. S. Brown, "Multi-view document rectification using boundary," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[23] H. I. Koo, J. Kim, and N. I. Cho, "Composition of a dewarped and enhanced document image from two view images," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1551–1562, 2009.

[24] S. You, Y. Matsushita, S. Sinha, Y. Bou, and K. Ikeuchi, "Multiview rectification of folded documents," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[25] C. L. Tan, L. Zhang, Z. Zhang, and T. Xia, "Restoring warped document images through 3d shape modeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 2, pp. 195–208, 2006.

[26] F. Courteille, A. Crouzil, J.-D. Durou, and P. Gurdjos, "Shape from shading for the digitization of curved documents," *Machine Vision and Applications*, vol. 18, no. 5, pp. 301–316, 2007.

[27] 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*, vol. 42, no. 11, pp. 2961–2978, 2009.

[28] Y. Takezawa, M. Hasegawa, and S. Tabbone, "Camera-captured document image perspective distortion correction using vanishing point detection based on radon transform," in *Pattern Recognition (ICPR), 2016 23rd International Conference on*. IEEE, 2016, pp. 3968–3974.

[29] B. Fu, M. Wu, R. Li, W. Li, Z. Xu, and C. Yang, "A model-based book dewarping method using text line detection," in *International Workshop on Camera Based Document Analysis and Recognition*, 2007, pp. 63–70.

[30] N. Stamatopoulos, B. Gatos, I. Pratikakis, and S. J. Perantonis, "A two-step dewarping of camera document images," in *International Workshop on Document Analysis Systems*, 2008, pp. 209–216.

[31] C. Liu, Y. Zhang, B. Wang, and X. Ding, "Restoring camera-captured distorted document images," *International Journal on Document Analysis and Recognition*, vol. 18, no. 2, pp. 111–124, 2015.

[32] B. Epshtein, "Determining document skew using inter-line spaces," in *IEEE International Conference on Document Analysis and Recognition*, 2011, pp. 27–31.

[33] S. N. Srihari and V. Govindaraju, "Analysis of textual images using the hough transform," *Machine Vision and Applications*, vol. 2, no. 3, pp. 141–153, 1989.

[34] I. Bar-Yosef, N. Hagbi, K. Kedem, and I. Dinstein, "Fast and accurate skew estimation based on distance transform," in *IEEE International Workshop on Document Analysis Systems*, 2008, pp. 402–407.

[35] C. Sun and D. Si, "Skew and slant correction for document images using gradient direction," in *IEEE International Conference on Document Analysis and Recognition*, vol. 1, 1997, pp. 142–146.

[36] P. Clark and M. Mirmehdi, "Rectifying perspective views of text in 3d scenes using vanishing points," *Pattern Recognition*, vol. 36, no. 11, pp. 2673–2686, 2003.

[37] J. Liang, D. DeMenthon, and D. Doermann, "Geometric rectification of camera-captured document images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 4, pp. 591–605, 2008.

[38] S. Messelodi and C. M. Modena, "Automatic identification and skew estimation of text lines in real scene images," *Pattern Recognition*, vol. 32, no. 5, pp. 791–810, 1999.

[39] G. K. Myers, R. C. Bolles, Q.-T. Luong, J. A. Herson, and H. B. Aradhye, "Rectification and recognition of text in 3-d scenes," *International Journal of Document Analysis and Recognition*, vol. 7, no. 2-3, pp. 147–158, 2005.

[40] M. Bušta, T. Drtina, D. Helekal, L. Neumann, and J. Matas, "Efficient character skew rectification in scene text images," in *Asian Conference on Computer Vision*, 2014, pp. 134–146.

[41] X. Zhang, Z. Lin, F. Sun, and Y. Ma, "Rectification of optical characters as transform invariant low-rank textures," in *IEEE International Conference on Document Analysis and Recognition*, 2013, pp. 393–397.

[42] ——, "Transform invariant text extraction," *The Visual Computer*, vol. 30, no. 4, pp. 401–415, 2014.

[43] B. Wang, C. Liu, and X. Ding, "A scheme for automatic text rectification in real scene images," in *SPIE/IS&T Electronic Imaging*, 2015, pp. 94 080M–94 080M.

[44] C. Merino-Gracia, M. Mirmehdi, J. Sigut, and J. L. González-Mora, "Fast perspective recovery of text in natural scenes," *Image and Vision Computing*, vol. 31, no. 10, pp. 714–724, 2013.

[45] B. Shi, X. Wang, P. Lyu, C. Yao, and X. Bai, "Robust scene text recognition with automatic rectification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4168–4176.

[46] F. Dellaert, S. M. Seitz, C. E. Thorpe, and S. Thrun, "Structure from motion without correspondence," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2000, pp. 557–564.

[47] P. Sturm and B. Triggs, "A factorization based algorithm for multi-image projective structure and motion," *European Conference on Computer Vision*, pp. 709–720, 1996.

[48] W. Hong, A. Y. Yang, K. Huang, and Y. Ma, "On symmetry and multiple-view geometry: Structure, pose, and calibration from a single image," *International Journal of Computer Vision*, vol. 60, no. 3, pp. 241–265, 2004.

[49] M. Park, K. Brocklehurst, R. T. Collins, and Y. Liu, "Deformed lattice detection in real-world images using mean-shift belief propagation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 10, pp. 1804–1816, 2009.

[50] H. I. Koo and N. I. Cho, "State estimation in a document image and its application in text block identification and text line extraction," in *European Conference on Computer Vision*, 2010, pp. 421–434.

[51] H. Lee, E. Shechtman, J. Wang, and S. Lee, "Automatic upright adjustment of photographs with robust camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 5, pp. 833–844, 2014.

[52] J. An, H. I. Koo, and N. I. Cho, "Rectification of planar targets using line segments," *Machine Vision and Applications*, vol. 28, no. 1, pp. 91–100, 2017.

[53] K. E. Atkinson, *An introduction to numerical analysis.* John Wiley & Sons, 2008.

[54] F. Shafait and T. M. Breuel, "Document image dewarping contest," in *Proceeding of International Workshop on Camera Based Document Analysis and Recognition*, 2007, pp. 181–188.

[55] B. Gatos, I. Pratikakis, and K. Ntirogiannis, "Segmentation based recovery of arbitrarily warped document images," in *Proceeding of International Conference Document Analysis and Recognition*, vol. 2, 2007, pp. 989–993.

[56] A. Masalovitch and L. Mestetskiy, "Usage of continuous skeletal image representation for document images de-warping," in *Proceeding of International Workshop on Camera Based Document Analysis and Recognition*, 2007, pp. 45–53.

[57] V. Levenstein, "Binary codes capable of correcting spurious insertions and deletions of ones," *Problems of Information Transmission*, vol. 1, no. 1, pp. 8–17, 1965.

[58] J. Ryu, H. I. Koo, and N. I. Cho, "Language-independent text-line extraction algorithm for handwritten documents," *IEEE Signal Processing Letters*, vol. 21, no. 9, pp. 1115–1119, 2014.

[59] Z. Lin, M. Chen, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *arXiv preprint arXiv:1009.5055*, 2010.

[60] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1083–1090.

[61] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. i Bigorda, S. R. Mestre, J. Mas, D. F. Mota, J. A. Almazan, and L. P. de las Heras, "ICDAR robust reading competition," in *IEEE International Conference on Document Analysis and Recognition*, 2013, pp. 1484–1493.

초록

카메라로 촬영한 텍스트 영상에 대해서, 광학 문자 인식(OCR)은 촬영된 장면을 분석하는데 있어서 매우 중요하다. 하지만 올바른 텍스트 영역 검출 후에도, 촬영한 영상에 대한 문자 인식은 여전히 어려운 문제로 여겨진다. 이는 종이의 구부러짐과 카메라 시점에 의한 기하학적인 왜곡 때문이고, 따라서 이러한 텍스트 영상에 대한 평활화는 문자 인식에 있어서 필수적인 전처리 과정으로 여겨진다. 이를 위한 왜곡된 촬영 영상을 정면 시점으로 복원하는 텍스트 영상 평활화 방법들은 활발히 연구되어지고 있다. 최근에는, 평활화가 잘 된 텍스트의 성질에 초점을 맞춘 연구들이 주로 진행되고 있다. 이러한 관점에서, 본 학위 논문은 텍스트 영상 평활화를 위하여 새로운 정렬 특성들을 다룬다. 이러한 정렬 특성들은 비용 함수로 설계되어지고, 비용 함수를 최소화하는 방법을 통해서 평활화에 사용되어지는 평활화 변수들이 구해진다. 본 학위 논문은 “문서 영상 평활화, 장면 텍스트 평활화, 일반 배경 속의 휘어진 표면 평활화”와 같이 3가지 세부 주제로 나눠진다.

첫 번째로, 본 학위 논문은 텍스트 라인들과 선분들의 정렬 특성에 기반의 문서 영상 평활화 방법을 제안한다. 기존의 텍스트 라인 기반의 문서 영상 평활화 방법들의 경우, 문서가 복잡한 레이아웃 형태이거나 적은 수의 텍스트 라인을 포함하고 있을 때 문제가 발생한다. 이는 문서에 텍스트 대신 그림, 그래프 혹은 표와 같은 영역이 많은 경우이다. 따라서 레이아웃에 강인한 문서 영상 평활화를 위하여 제안하는 방법은 정렬된 텍스트 라인뿐만 아니라 선분들도 이용한다. 올바르게 평활화 된 선분들은 여전히 일직선의

형태이고, 대부분 가로 혹은 세로 방향으로 정렬되어 있다는 가정 및 관측에 근거하여, 제안하는 방법은 이러한 성질들을 수식화하고 이를 텍스트 라인 기반의 비용 함수와 결합한다. 그리고 비용 함수를 최소화 하는 방법을 통해, 제안하는 방법은 종이의 구부러짐, 카메라 시점, 초점 거리와 같은 평활화 변수들을 추정한다. 또한, 오검출된 텍스트 라인들과 임의의 방향을 가지는 선분들과 같은 이상점(outlier)을 고려하여, 제안하는 방법은 반복적인 단계로 설계된다. 각 단계에서, 정렬 특성을 만족하지 않는 이상점들은 제거되고, 제거되지 않은 텍스트 라인 및 선분들만이 비용함수 최적화에 이용된다. 수행한 실험 결과들은 제안하는 방법이 다양한 레이아웃에 대하여 강인함을 보여준다.

두 번째로는, 본 논문은 장면 텍스트 평활화 방법을 제안한다. 기존 장면 텍스트 평활화 방법들의 경우, 가로/세로 방향의 획, 대칭 형태와 같은 문자가 가지는 고유의 생김새에 관련된 특성을 이용한다. 하지만, 이러한 방법들은 문자들의 정렬 형태는 고려하지 않고, 각각 개별 문자에 대한 특성들만을 이용하기 때문에 여러 문자들로 구성된 텍스트에 대해서 잘 정렬되지 않은 결과를 출력한다. 이러한 문제점을 해결하기 위하여, 제안하는 방법은 문자들의 정렬 정보를 이용한다. 정확하게는, 문자 고유의 모양뿐만 아니라 정렬 특성들도 함께 비용함수로 수식화되고, 비용함수를 최소화하는 방법을 통해서 평활화가 진행된다. 또한, 문자들의 정렬 특성을 수식화하기 위하여, 제안하는 방법은 텍스트를 각각 개별 문자들로 분리하는 문자 분리 또한 수행한다. 그 뒤, 텍스트의 위, 아래 선들을 RANSAC 알고리즘을 이용한 최소 제곱법을 통해 추정한다. 즉, 전체 알고리즘은 문자 분리와 선 추정, 평활화가 반복적으로 수행된다. 제안하는 비용함수는 볼록(convex)형태가 아니고 또한 많은 변수들을 포함하고 있기 때문에, 이를 최적화하기 위하여 Augmented Lagrange Multiplier 방법을 이용한다. 제안하는 방법은 일반 촬영 영상과 합성된 텍스트 영상을 통해 실험이 진행되었고, 실험 결과들은 제안하는 방법이 기존 방법들에 비하여 높은 인식 성능을 보이면서 동시에 시각적으로도 좋은 결과를 보임을 보여준다.

마지막으로, 제안하는 방법은 일반 배경 속의 휘어진 표면 평활화 방법으로도 확장된

다. 일반 배경에 대해서, 약병이나 음료수 캔과 같이 원통 형태의 물체는 많이 존재한다. 그들의 표면은 일반 원통 표면(GCS)으로 모델링이 가능하다. 이러한 휘어진 표면들은 많은 문자와 그림들을 포함하고 있지만, 포함된 문자는 문서에 비해서 매우 불규칙적인 구조를 가지고 있다. 따라서 기존의 문서 영상 평활화 방법들로는 일반 배경 속 휘어진 표면 영상을 평활화하기 힘들다. 많은 휘어진 표면은 잘 정렬된 선분들 (테두리 선 혹은 바코드)을 포함하고 있다는 관측에 근거하여, 제안하는 방법은 앞서 제안한 선분들에 대한 함수를 이용하여 휘어진 표면을 평활화한다. 다양한 둥근 물체의 휘어진 표면 영상들에 대한 실험 결과들은 제안하는 방법이 평활화를 정확하게 수행함을 보여준다.

주요어: 문서 영상 평활화, 장면 텍스트 평활화, 문자 분리, 휘어진 표면 평활화

학 번: 2013-30220