Available online at www.sciencedirect.com
SCIENCE DIRECT
Image and Vision Computing 23 (2005) 541–553
image AND vision COMPUTING
www.elsevier.com/locate/imavis

Perspective rectification of document images using fuzzy set and morphological operations

Shijian Lu, Ben M. Chen*, C.C. Ko

Electrical and Computer Engineering Department, National University of Singapore, Singapore 119260, Singapore

Received 13 May 2004; received in revised form 15 October 2004; accepted 1 January 2005

Abstract

In this paper, we deal with the problem of document image rectification from image captured by digital cameras. The improvement on the resolution of digital camera sensors has brought more and more applications for non-contact text capture. Unfortunately, perspective distortion in the resulting image makes it hard to properly identify the contents of the captured text using traditional optical character recognition (OCR) systems. We propose in this work a new technique, which is capable of removing perspective distortion and recovering the fronto-parallel view of text with a single image. Different from reported approaches in the literature, the image rectification is carried out using character stroke boundaries and tip points (SBTP), which are extracted from character strokes based on multiple fuzzy sets and morphological operators. The algorithm needs neither high-contrast document boundary (HDB) nor paragraph formatting (PF) information. Experimental results show that our rectification process is fast and robust.

Keywords: Document image analysis; Document image rectification; Optical character recognition; Morphological image processing; Fuzzy sets

1. Introduction

Document scanner is widely used to capture text and transform it into electric form for further processing. As camera resolution rises in recent years, high-speed non-contact text capture through digital cameras is becoming an alternative choice. Unfortunately, perspective distortion coupled with captured images by digital camera brings up a new problem to the traditional optical character recognition (OCR) system. Similar to the skew compensation operation required after the scanning process, perspective distortion must be removed before the document image is fed to the OCR system.

A proliferation of research has been dedicated to the detection and correction of document skew that is introduced during the scanning process. The proposed correction approaches can be roughly classified into three categories, namely, Hough transform based method [1,2], projection profile based method [3], and nearest neighbor based method [4]. One common feature of these methods is that they all assume that skew distortion is rotation-induced, which means that the top line and base line of each text line are still parallel to each other within the scanned document image. Therefore, none of them can handle the perspectively distorted document images where the parallel relation between top line and base line of text lines is totally destroyed.

Although the geometry of rectification is fairly mature [5], few rectification techniques have been reported in the literature for perspectively distorted document images captured through digital cameras. In [6], the quadrilaterals formed by the boundary between the background and plane where text lies are utilized to get a fronto-parallel view of perspectively distorted text. After the extraction of quadrilaterals using the perceptual grouping method, a bilinear interpolation operation is implemented to construct the corrected document image. As the algorithm depends heavily on the extraction of quadrilateral, the existence of the high-contrast document boundary (HDB) within the captured document image is a must for correct rectification.

Instead of using document boundaries that do not always exist in real scene, Pilu proposed a new rectification

* Corresponding author. Tel.: +65 6874 2289; fax: +65 6779 1103.
E-mail address: bmchen@nus.edu.sg (B.M. Chen).

0262-8856/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2005.01.003

approach in [7] based on the extraction of illusory clues. To extract the horizontal clues, the character or group of characters is transformed into blob first and a pairwise saliency measure is computed for pairs of neighboring blobs, which indicates how likely they belong to one text line. After that, a network based on perceptual organization principles is transversed over the text and horizontal clues are calculated as the salient linear groups of blobs. Though working well on the extraction of horizontal clues, the method cannot extract enough vertical information.

In Dance [8], distorted document image is rectified using two principal vanishing points, which are estimated based on the parallel lines extracted from the text lines and the vertical paragraph margins (VPM). The main drawback of this approach is that it works only on fully aligned text as it relies heavily on the existence of VPM features. Besides, the means to extract the parallel lines is not clarified either.

In [9], Clark estimates two vanishing points based on some paragraph formatting (PF) information. More specifically, the horizontal vanishing point is calculated based on a novel extension of 2D projection profile and the vertical vanishing point based on some PF information such as VPM or text line spacing variation when paragraphs are not fully aligned. Well-formatted paragraphs are required to implement such a rectification method.

In this paper, we propose a new document image rectification technique that is able to detect and correct the perspective distortion resulting from the camera capturing process within the three-dimension space. Different from reported rectification methods that depend heavily on the HDB or PF information, the proposed technique uses the stroke boundaries and tip points (SBTP) as shown in Fig. 1 to rectify the document images when HDB or PF information is not available. The use of SBTP has multiple advantages. Firstly, unlike the HDB that does not always exist in real scene, SBTP are extracted directly from character strokes. Therefore, the proposed approach based on SBTP can rectify the document images without HDB features. Secondly, it can rectify document images that contain only a single text line or even a single word, which cannot be handled by approaches using either HDB or PF information. Lastly, with a little adaptation, the proposed rectification method can be extended to rectify the geometrically distorted document images where text lies on curved instead of plane surface.

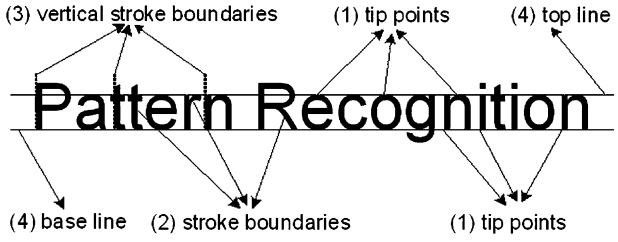

Fig. 1. Text line definition.

In our proposed approach, the distortion rectification is implemented based on the construction of quadrilateral correspondence. The SBTP features shown in Fig. 1, namely, the features labeled with (1) and (2), are first extracted using some morphological operators. Then, three fuzzy sets that characterize size, linearity, and pose properties of extracted stroke boundaries are constructed. The specific stroke boundaries that indicate the vertical direction shown by (3) in Fig. 1 are identified using a fuzzy aggregation operator. To extract the horizontal clues, the tip points are first classified using a novel point tracing technique that utilizes the point to point and point to line distance constraints. Then, horizontal lines corresponding to the base line and top line of text line shown as (4) in Fig. 1 are fitted using the well known least square method (see, for example, [10]).

The source quadrilaterals are constructed based on two sets of orthogonal lines, which are fitted from the identified vertical stroke boundaries and classified tip points. Then, characters within the source quadrilaterals are counted using the connected component analysis of Jain et al. [11] and the target quadrilaterals corresponding to the source quadrilaterals are built based on the counted character number and the approximation that the height width ratio of character with no ascendant and descendant is 1:1.

With multiple quadrilateral correspondences constructed, multiple rectification homographies can be estimated using the four-point algorithm reported in Hartley and Zisserman [12]. An optimal one is selected based on the facts that classified tip points should lie on multiple horizontal lines and vertical boundaries should be rectified to multiple vertical line segments in the rectified document image. With the optimal homography, the distorted document image can be rectified at last. The whole rectification process is summarized in Fig. 2.

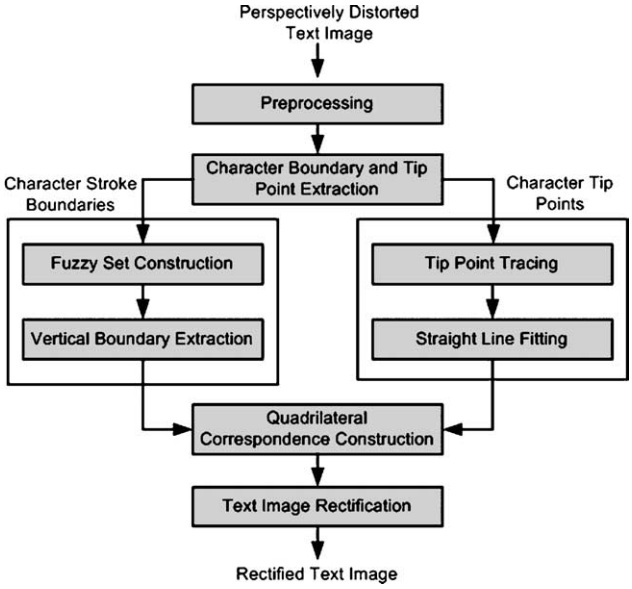

Fig. 2. Document image rectification process.

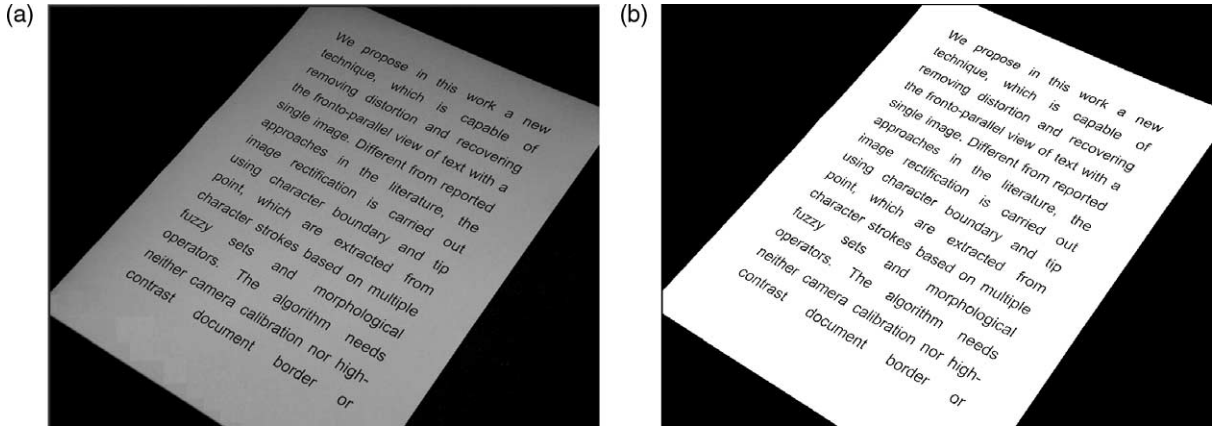

Fig. 3. Source image: (a) distorted image; (b) binarized image.

The outline of the paper is as follows: In Section 2, we present the details of the proposed rectification technique based on SBTP extracted using fuzzy set and morphological operations. Experiment results together with detailed comparison to other approaches will then be given in Section 3. Finally, we draw some concluding remarks in Section 4.

2. Proposed rectification technique

We present in this section the detailed procedure of the proposed rectification technique. In particular, we will divide this section into a few subsections, which deal with the extraction of SBTP features, the feature processing steps, including the tip point tracing process and vertical boundary identification process, and the document image rectification.

2.1. Feature extraction

To facilitate SBTP extraction, we first transform the document image into a binary representation. A lot of image binarization methods have been proposed in the literature (see, for example, [13–16]). In most cases, document image binarization is not a tough task since text and background are generally designed to be highly contrasted. In our proposed approach, we adopt the global thresholding technique [16] to binarize the captured document image. Fig. 3(a) shows the distorted document image captured through a digital camera and Fig. 3(b) shows its binarized form. This example will be used throughout this section to illustrate our rectification technique.

2.1.1. Extraction of tip points

Mathematical morphology [17] is a powerful tool for extracting image components that are useful in the representation and description of region shape such as boundaries, skeletons, and convex hull. In the proposed approach, we extract tip points using some customized morphological operators. Fig. 4(a)–(d) show four sets of customized structure elements, where each set consists of five structure elements. The gray pixels define the neighborhood and the darker pixels give the origins of the customized structure elements. The use of multiple structure elements in each set is to get rid of some unwanted points that deviate significantly from the top or base line of text lines.

As shown in Fig. 4, with any set of the structure elements, we will be able to obtain the other three by rotating it with an angle of 90, 180 and 270°. To extract the tip points near the top line and base line positions, we use two combinations of the four structure element sets with each combination containing two groups and each group containing two structure element sets. At the same time, two sets within each group must be orthogonal to each other, i.e. one must be a 90° rotation of the other. For example, for the four sets given in Fig. 4, if we choose one combination as {(a), (b)} and {(c), (d)}, the other combination has to be chosen as {(a), (d)} and {(b), (c)}.

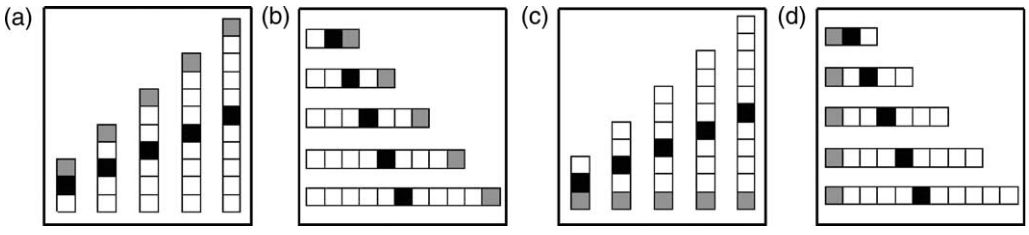

Fig. 4. Character tip point extraction: four sets of structure elements.

For the binarized document image shown in Fig. 3(b), the combination {(a), (b)} and {(c), (d)} can be used to extract the desired tip points. The group {(a), (b)} can be used to extract the tip points near the top line position and the group {(c), (d)} can be utilized to extract the tip points near the base line position. The related morphological operations can be summarized as follows:

$$BDY_t = \{[I \underline{\vee} (I \Theta A_1)] \cap \cdots \cap [I \underline{\vee} (I \Theta A_5)]\} \cap \{[I \underline{\vee} (I \Theta B_1)] \cap \cdots \cap [I \underline{\vee} (I \Theta B_5)]\}, \quad (1)$$

and

$$BDY_t = \{[I \underline{\vee} (I \Theta C_1)] \cap \cdots \cap [I \underline{\vee} (I \Theta C_5)]\} \cap \{[I \underline{\vee} (I \Theta D_1)] \cap \cdots \cap [I \underline{\vee} (I \Theta D_5)]\}, \quad (2)$$

where I refers to the document image shown in Fig. 3(b) and A_i, B_i, C_i, D_i, $i=1, 2,\ldots, 5$, refer to the four sets of the structure elements as in Fig. 4(a)–(d), respectively. The symbol Θ represents to the erosion operation, and the symbol $\underline{\vee}$ represents the XOR operation. Fig. 5 shows the extracted tip point at the base line position.

We note that the image in Fig. 3 shows only one type of perspective distortion, in which the text lines go roughly from the top left to the bottom right corner. For this distortion, in which the text lines go from the bottom left to the top right corner, the combinations {(a), (d)} and {(b), (c)} can be employed to extract the desired tip points.

For newly captured document images, a rough orientation of the text lines can be determined based on the relative position of adjacent characters. For each character candidate represented with its centroid, the nearest character to it can be searched and this searched one generally lies on the left or right of the studied character candidate within the same text line. The relative position of these two adjacent characters, which is normally close to the orientation of text line, can thus be calculated as the slope of straight line that passes their centroids. With a large number of character candidates studied in this way, the rough orientation of text line can be determined as the average slope of calculated straight lines.

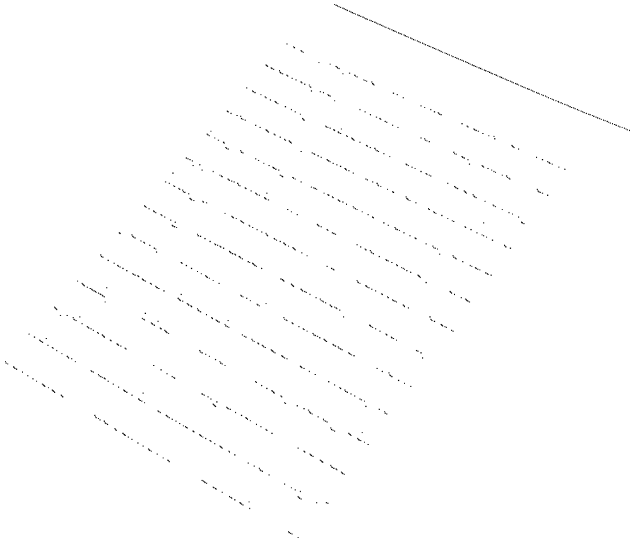

Fig. 5. Extracted tip points at the base line position.

2.1.2. Extraction of stroke boundaries

In our proposed approach, we use some customized structure elements to extract character boundaries. The boundary extraction is to identify lines that indicate the vertical direction of text. We divide the extraction process into two steps, corresponding to extracting boundaries on the left and right side of character strokes.

Fig. 6(a)–(d) show four sets of structure elements with each set consisting of three structure elements. Similar to the structure elements used for the tip point extraction, the four sets of structure elements here satisfy an orthogonal relationship, i.e. all of them are related by an angle of 90, 180 or 270°. Any two sets that differ by 180° can be combined to extract boundaries on the two sides of character strokes.

For the distorted document image shown in Fig. 3, where the text lines go from the top left to the bottom right corner, structure element sets (b) and its counterpart (d) shown in Fig. 6 can be used together to extract the left side and right side stroke boundaries of the text shown in Fig. 3(b). The related morphological operations can be summarized as follows:

$$BDY_l = [I \underline{\vee} (I \Theta B_1)] \cap \{[I \underline{\vee} (I \Theta B_2)] \cup [I \underline{\vee} (I \Theta B_3)]\}, \quad (3)$$

and

$$BDY_r = [I \underline{\vee} (I \Theta D_1)] \cap \{[I \underline{\vee} (I \Theta D_2)] \cup [I \underline{\vee} (I \Theta D_3)]\}, \quad (4)$$

where I refers to the document image given in Fig. 3(b), and B_i, and D_i, $i=1, 2, 3$, represent, respectively, the two sets of the structure elements shown in Fig. 6(b) and (d).

Similarly, the pair (a) and (c) can be combined to extract the stroke boundaries from document images with text lines running from the bottom left to the top right corner. The determination of text line orientation has been discussed in

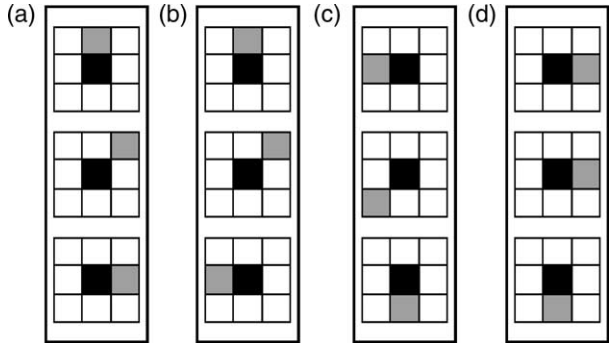

Fig. 6. Character stroke boundary extraction: four sets of structure elements.

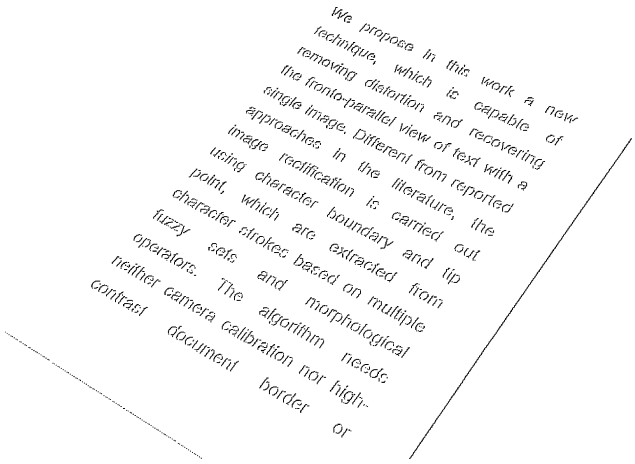

Fig. 7. Extracted left side stroke boundaries.

Section 2.1.1. Fig. 7 shows the extracted stroke boundaries on the left side of the text shown in Fig. 3.

2.2. Feature processing

In the following, we present the processing of the extracted features of the distorted document images. We aim to identify two groups of straight lines representing the horizontal and vertical directions of the rectified document image.

2.2.1. Tip point classification

For the distorted document image of Fig. 3, most of extracted tip points shown in Fig. 5 lie roughly on multiple straight lines that correspond to the base lines of the text lines. In the following, we propose a point tracing technique to classify the extracted tip point. Ideally, each classified point group should fit into a desired top line or base line of the text lines.

Fig. 8 shows the tracing process where the white, light gray and darker gray points correspond to the extracted tip points from three text lines. In our proposed approach, we use both point to point and point to line distance constraints to classify the tip points. For example, in Fig. 8, Points 1–7 are seven points classified to belong to the same group. Line L is fitted based on all classified tip points, which include points 1–7 in this example. The target of the tip point tracing process is to search for the next point candidate that is nearest to the last classified point (point 7) and at the same time satisfies two distance constraints.

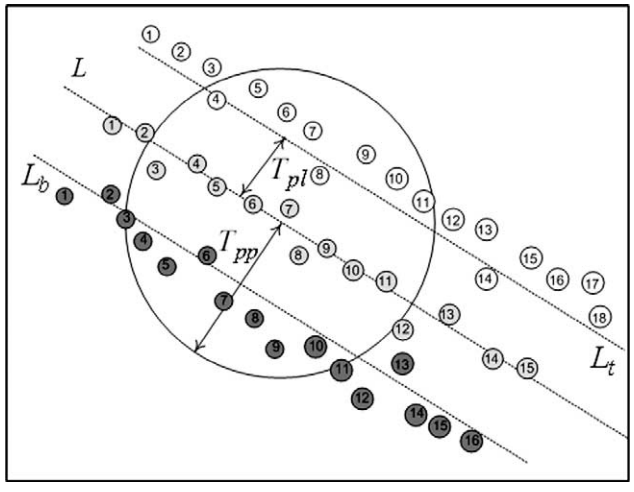

Fig. 8. Tip point tracing process.

In Fig. 8, parameter T_{pp} refers to point to point distance threshold. All tip points that satisfy this constraint must fall within the circle defined by the last classified point, i.e. Point 7 in this example with its position being the center and T_{pp} being its radius. Parameter T_{pl} refers to the point to line distance threshold. Accordingly, all tip points that satisfy this second constraint must lie within the band defined by two parallel lines L_b and L_t, which are parallel and with the same distance, T_{pl} to Line L. As shown in Fig. 8, the nearest point to the last classified point, Point 7, satisfying the above two distance constraints is Point 8.

The tracing algorithm is summarized as follows:

Inputs:
  All extracted tip point *TP*
  Point to point distance threshold *P_PThre*
  Point to line distance threshold *P_LThre*

Procedure TPC (TP, P_PThre, P_LThre)

(1) Initialize $i=1$
(2) Construct source vector *SV* and initialize it with all extracted tip points *TP*
(3) Repeat
(4) Construct a new target vector TV_i and remove the first point TP_1 of *SV* to TV_i
(5) Shift all left elements in *SV* one step forwards
(6) Search over *SV* for the point that is nearest to first element TP_1 and remove it to TV_i
(7) Shift all elements after the last removed point one step forwards
(8) Repeat
(9) Fit the line using all tip points in TV_i
(10) Search over *SV* and remove the one that is nearest to last removed point and satisfies two constraints defined by *P_PThre* and *P_LThre* to TV_i
(11) Shift all elements after the last removed point one step forwards
(12) $i=i+1$
(13) Until no match points in *SV*
(14) Until *SV* is null
(15) Return *TV*

We note that the two thresholds T_{pp} and T_{pl} must be chosen with care. For the point to point distance threshold, T_{pp}, should be large enough to span the blank between words. While for the point to line distance threshold, T_{pl}, should be small enough to exclude tip points that belong to other text lines. In our proposed approach, the two

thresholds are defined as follows:

$$T_{\mathrm{pl}} = \mathrm{VSB}_{\mathrm{avg}} \text{ and } T_{\mathrm{pp}} = k_p \cdot \mathrm{VSB}_{\mathrm{avg}} \tag{5}$$

where $\mathrm{VSB}_{\mathrm{avg}}$ represents the average size of identified vertical stroke boundaries, which will be determined in the following section. Parameter k_{p} is used to adjust the point to point threshold and it normally ranges from 3 to 5.

Finally, we note that the number of classified point groups is not same but more than the number of top lines and base lines. Such a mismatch results mainly from some characters such as 'b', 'q' and 'g', whose tip points deviate pretty far from the top line or base line. As such, we need to employ a size filter on the classified point groups to further identify the desired point groups. In our proposed approach, the threshold used in the size filter is determined as:

$$T_s = k_s \cdot \mathrm{Size}_{\max} \tag{6}$$

where $\mathrm{Size}_{\max}$ is the size of the point group whose size is biggest among all classified point groups. Parameter k_{s} can generally be determined as a scalar between 0.4 and 0.6.

For the document image given in Figs. 3(b) and 9 shows the fitted top lines and base lines based on the classified tip points. It is evident that one top line is filtered out by the proposed size filter. Under such a circumstance, the corresponding base line counterpart becomes useless, but we are still able to use the remaining top line and base line pairs for the estimation of rectification homography.

2.2.2. Vertical stroke boundary identification

We now proceed to propose a fuzzy technique to identify the stroke boundaries that indicate the vertical direction of text. In our approach, the connected component analysis is first implemented and the extracted stroke boundaries are labeled as multiple foreground regions, where the regions that have small size, say, less than 5 pixels, can be firstly filtered out because they cannot be the desired vertical stroke boundaries. Then, three fuzzy sets that characterize the size, the orientation and the linearity properties of the left foreground regions are defined. Lastly, the vertical stroke boundaries are identified with an aggregation operation over three constructed fuzzy sets.

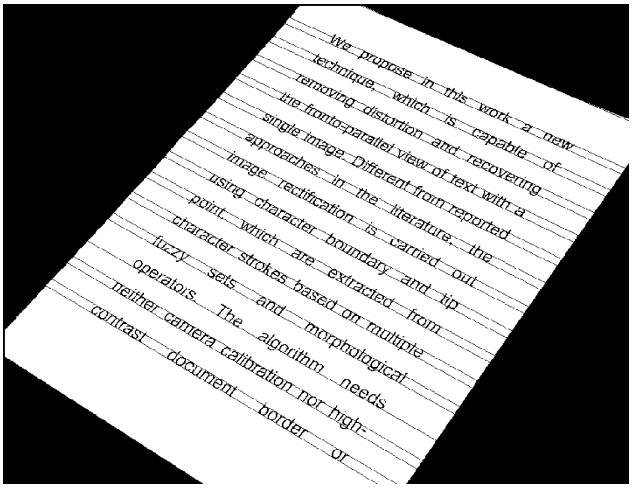

Fig. 9. Fitted top lines and base lines.

Three measures are implemented upon the extracted foreground regions. The first measure is the region size defined as the pixel number of the foreground region. The second measure is the region orientation that is estimated as the angel between the fitted lines based on region pixels and the positive x axis. The last measure is the region linearity that describes the 'straightness' of a region. We define the linearity measure as:

$$\mathrm{dist} = \frac{1}{n}\sum_{i=1}^{n} d(p_i, l), \tag{7}$$

where n is the pixel number of a region, l refers to the fitted line based on the region pixels using the least square method, and function d calculates the distance between pixel p_i and the fitted line.

As the foreground regions that correspond to the vertical stroke boundaries generally have larger size compared with other ones, we choose the well known Zadeh's S-function $S(x; a, b, c)$ (see, e.g. [18]):

$$S(x;a,b,c) = \begin{cases} 0 & x \le a \\ 2\left(\frac{x-a}{c-a}\right)^2 & a < x \le b \\ 1 - 2\left(\frac{x-c}{c-a}\right)^2 & b < x \le c \\ 1 & x > c \end{cases}, \tag{8}$$

as a membership function to describe the size property. Fig. 10(a) shows the S-function. To identify the vertical stroke boundaries, parameters a and c are determined as the minimal size and average size of all extracted regions and parameter b represents crossover point.

For the linearity measure, the normalized distance calculated in Eq. (7) is used to describe the straightness property of extracted foreground regions. The smaller the calculated distance is, the straighter the foreground region. As vertical stroke boundaries are generally more straight compared with other stroke boundaries, we choose the complement of the S-function $C(x; a, b, c) = 1 - S(x; a, b, c)$ as a membership function to describe the linearity property. Parameters a and c are determined as the minimal and average distance calculated using Eq. (7) and parameter b represents crossover point.

For a perspectively distorted document image, the orientation of all desired vertical boundaries generally varies within a small range. At the same time, the orientation of most of extracted stroke boundaries as shown in Fig. 7 is normally close to that of desired vertical stroke boundaries. Therefore, we calculate the average orientation of the extracted boundaries and the boundary with orientation closer to the average orientation should be assigned a higher membership value. For each extracted stoke boundary, its orientation can be estimated as the slope

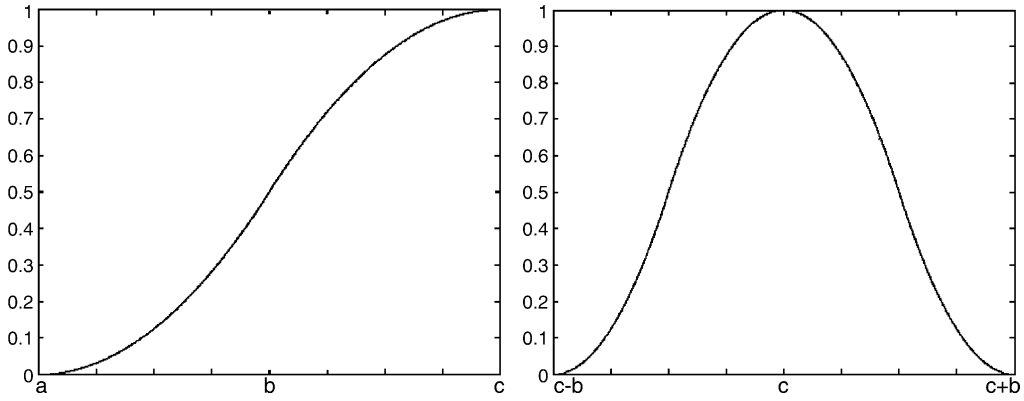

Fig. 10. Membership function: (a) S-function; (b) bell-shaped function.

of straight line fitted based on boundary pixels using least square regression method. The average orientation of extracted stroke boundaries can thus be estimated as the mean or median of the slope of all fitted straight lines.

In our proposed approach, we choose a bell-shaped function $\pi(x; b, c)$:

$$\pi(x;b,c) = \begin{cases} S(x; c-b, c-b/2, c) & x \le c \\ 1 - S(x; c, c+b/2, c+b) & x > c \end{cases}, \tag{9}$$

to describe the orientation property of extracted foreground regions. The symbol S denotes the Zadeh's S-function defined in Eq. (8). Fig. 10(b) shows the bell-shaped function. Parameter c is taken as the average region orientation and parameter b can be obtained based on the calculated average orientation.

With three defined measures, each isolated boundary can be represented as a triple that is composed of size, linearity and orientation membership value. To facilitate vertical boundary identification, an aggregation operator is employed to transform three fuzzy sets into one. In our proposed approach we adopt the compensatory operators defined by Zimmermann and Zysno [19], in which the aggregation operation arises as a sort of combination of 'pure' logical AND and OR connectives. This provides the required compensation mechanism. For two fuzzy sets A and B, the aggregation set is defined as:

$$(A\Theta B)(x) = [(A \cap B)(x)]^{1-\gamma} + [(A \cup B)(x)]^{\gamma}, \tag{10}$$

where $\cup$ and $\cap$ are union and intersection operations, and γ stands for the compensation factor $\gamma \in [0,1]$, indicating where the actual operator is located between AND and OR. The aggregation set that combines three constructed sets is thus:

$$(P\Theta L\Theta S)(x) = [(P \cap (L\Theta S)(x))(x)]^{1-\beta} + [(P \cup (L\Theta S)(x))(x)]^{\beta}, \tag{11}$$

where S, L and P are the size, linearity and orientation sets. $(L\Theta S)(x)$ refers to the aggregation operation on the linearity and size sets. After the combination of three fuzzy set, vertical boundaries can be identified as the α cut of aggregation set. For the document image shown in Figs. 3(b) and 11 shows the identified vertical stroke boundaries on both sides of character strokes.

We note that parameter α must be chosen with care. If parameter α is too small, some false boundaries may be identified. Otherwise, some truly vertical boundaries may be excluded. Fig. 12 shows vertical boundary identification results with respect to parameter α. The * curve represents the number of correctly identified vertical boundaries with respect to parameter α, whereas the x curve shows the number of wrongly identified vertical boundaries. We determine the parameter α based on the desired number of vertical boundaries, which is closely related to the number of characters within the document image.

2.3. Perspective distortion rectification

We utilize quadrilateral correspondence information to rectify the distorted document image. For each constructed quadrilateral correspondence, a rectification homography can be estimated and the distorted document image is

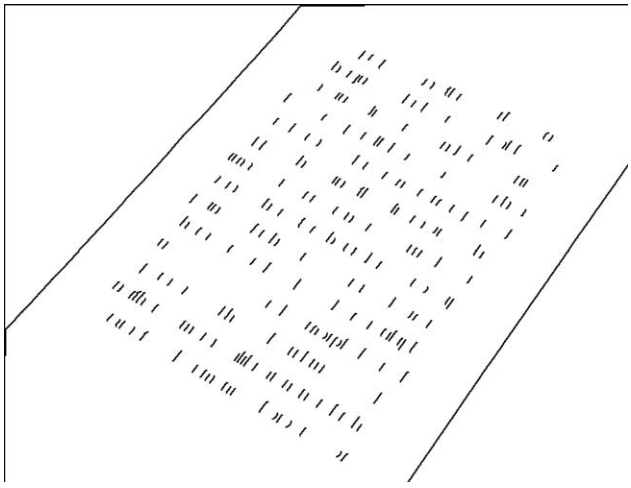

Fig. 11. Identified vertical boundaries.

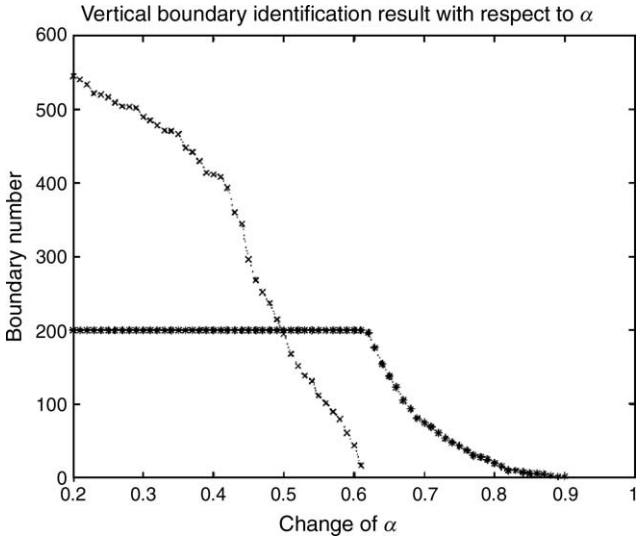

Fig. 12. Vertical boundary identification with respect to α.

rectified based on the optimal homography. The construction of source and target quadrilaterals, the estimation of optimal homography, as well as the final document image rectification will be discussed in this section.

We construct the source quadrilaterals using the identified vertical boundaries and classified tip points. Generally the quadrilateral vertexes can be chosen as the endpoints of vertical boundaries. But for the characters with ascendant or descendant or both, the feature points may deviate quite a lot from the base and top line. Therefore, we fit straight lines based on identified vertical stroke boundaries shown in Fig. 11 and determine the quadrilateral vertexes as intersection of the fitted vertical and horizontal lines that belong to the same text line. Fig. 13(a) shows one constructed source quadrilateral.

Since perspective capturing process impairs the geometric relation between lines, we construct the target quadrilateral based on the character number within source quadrilateral and the approximation that the height width ratio of character with no ascendant and descendant is 1:1. It should be clarified that the character width height ratio 1:1 here is only an average approximation as there may exist a large difference on the width height ratio for different characters such as 'm' and 'i'. To make this approximation more close to the fact, the two vertical stroke boundaries chosen to construct the source quadrilateral must be far enough to each other so that the constructed source quadrilateral is able to enclose more characters.

Characters within source quadrilateral are first counted based on connected component analysis. The blank between words is taken as one character, which can be detected based on the distance between the centroid of two adjacent characters. With the approximated character height width ratio, i.e. 1:1, the relation between the height and width of target quadrilaterals can be restored as

$$l_q = n \cdot h_q, \tag{12}$$

The parameters l_q and h_q are the length and height of the target quadrilateral. n is the number of character within the source quadrilateral, including the blanks between words.

With each source quadrilateral, one target quadrilateral must be constructed to estimate the rectification homography. The height of the target quadrilateral, which is equal to the height of rectified text lines, can be uniformly specified based on the desired text size within the rectified document image. With the character width height ratio estimated in Eq. (12), the length of the target quadrilateral can be determined accordingly. The position of the target quadrilateral, which is same to the position of the top left corner of target quadrilateral, can be estimated based on the position of the corresponding source quadrilateral within the captured document image. More specifically, the horizontal position of the target quadrilateral can be determined based on the number of characters on the left of source quadrilateral within the same text line. The vertical position of the target quadrilateral can be estimated based on the position of text line where source quadrilateral lies, which can be determined based on the relative position of intersections between the fitted top lines and straight line $x=0$. Fig. 13(b) show one constructed target quadrilateral corresponding to source quadrilateral shown in Fig. 13(a).

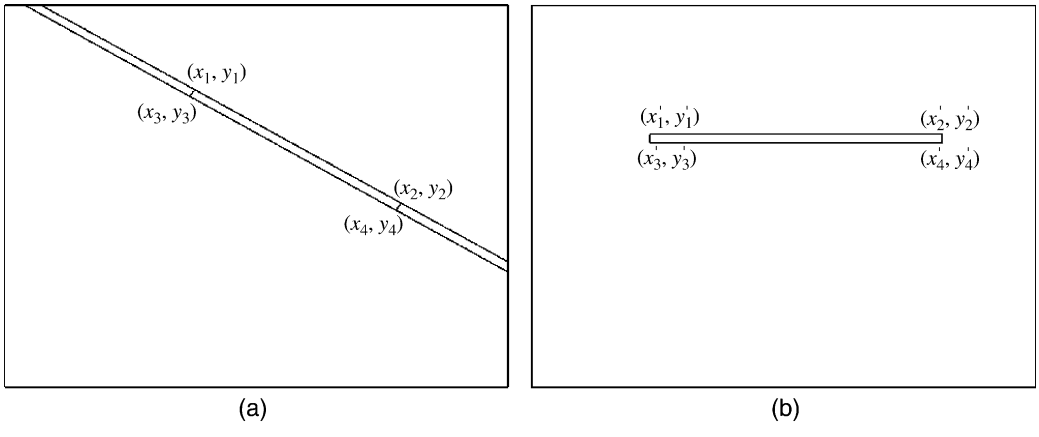

Fig. 13. One constructed quadrilateral correspondence: (a) the source quadrilateral; (b) the target quadrilateral.

With multiple pairs of source and target quadrilaterals, multiple rectification homographies can be calculated using the four point algorithm [12]. With four point correspondences the homography between distorted view and front-parallel view of document image can be estimated as follows:

$$H = A^{-1} \cdot R \tag{13}$$

where H is the homography matrix and matrixes A, R are constructed using four point correspondences. The three matrixes take the following forms:

$$H = \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix}, \quad A = \begin{bmatrix} -x_1 & -y_1 & -1 & 0 & 0 & 0 & x'_1 \cdot x_1 & x'_1 \cdot y_1 \\ 0 & 0 & 0 & -x_1 & -y_1 & -1 & y'_1 \cdot x_1 & y'_1 \cdot y_1 \\ -x_2 & -y_2 & -1 & 0 & 0 & 0 & x'_2 \cdot x_2 & x'_2 \cdot y_2 \\ 0 & 0 & 0 & -x_2 & -y_2 & -1 & y'_2 \cdot x_2 & y'_2 \cdot y_2 \\ -x_3 & -y_3 & -1 & 0 & 0 & 0 & x'_3 \cdot x_3 & x'_3 \cdot y_3 \\ 0 & 0 & 0 & -x_3 & -y_3 & -1 & y'_3 \cdot x_3 & y'_3 \cdot y_3 \\ -x_4 & -y_4 & -1 & 0 & 0 & 0 & x'_4 \cdot x_4 & x'_4 \cdot y_4 \\ 0 & 0 & 0 & -x_4 & -y_4 & -1 & y'_4 \cdot x_4 & y'_4 \cdot y_4 \end{bmatrix}, \quad R = \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ x'_3 \\ y'_3 \\ x'_4 \\ y'_4 \end{bmatrix}, \tag{14}$$

where the 3×3 homography matrix is expressed in vector form and h_{33} is equal to 1 under homogeneous frame. Four point correspondences $\langle (x_i, y_i), (x'_i, y'_i) \rangle$, $i = 1, \ldots, 4$, are taken as the four vertex of the constructed source and target quadrilaterals as shown in Fig. 13.

As both feature extraction and feature processing processes introduce errors, the estimated location of source quadrilateral contains errors as well. A criterion must be set to choose the homography that optimize the rectification performance. Based on the facts that tip points lie ideally on multiple horizontal lines and vertical boundaries lie on multiple vertical lines within the rectified document image, we define the objection function as

$$J = \frac{1}{m} \sum_{i=1}^{m} \mathrm{abs}\left(\frac{S_{l_i}}{S_{\mathrm{avg}}} \right) + \frac{1}{n} \sum_{j=1}^{n} \mathrm{abs}\left(\frac{ptx_j - pbx_j}{\mathrm{Dist}_{\mathrm{avg}}} \right), \tag{15}$$

where m is the number of horizontal lines and n is the number of vertical boundaries identified. S_{l_i} is the orientation of *i-th* rectified horizontal line and S_{avg} is the orientation average. ptx_j and pbx_j represent two horizontal coordinates of vertex of j-th rectified vertical boundary and

$$\mathrm{abs}\left(\frac{\mathrm{ptx}_j - \mathrm{pbx}_j}{\mathrm{Dist}_{\mathrm{avg}}} \right) \tag{16}$$

is to the normalized distance in horizontal direction between vertexes of that boundary.

The first part on the right side of Eq. (15) represents the sum of normalized orientation of the rectified horizontal lines, which ideally should be zero, and the second part refers to the sum of normalized vertex distance of rectified vertical stroke boundaries in horizontal direction, which ideally should be zero as well.

The optimal homography is accordingly determined as the one that minimizes the objection function defined in Eq. (15). Fig. 14 shows the rectified document image based on the estimated optimal homography.

3. Experiment results

We have tested a large number of document images captured using different digital cameras. Experiment results show that our proposed rectification approach can deal with images captured from different orientations and distances. We implement our algorithm in C++ and the computer is equipped with a Pentium 4 CPU and 512M memory. At the present stage the average rectification process takes around

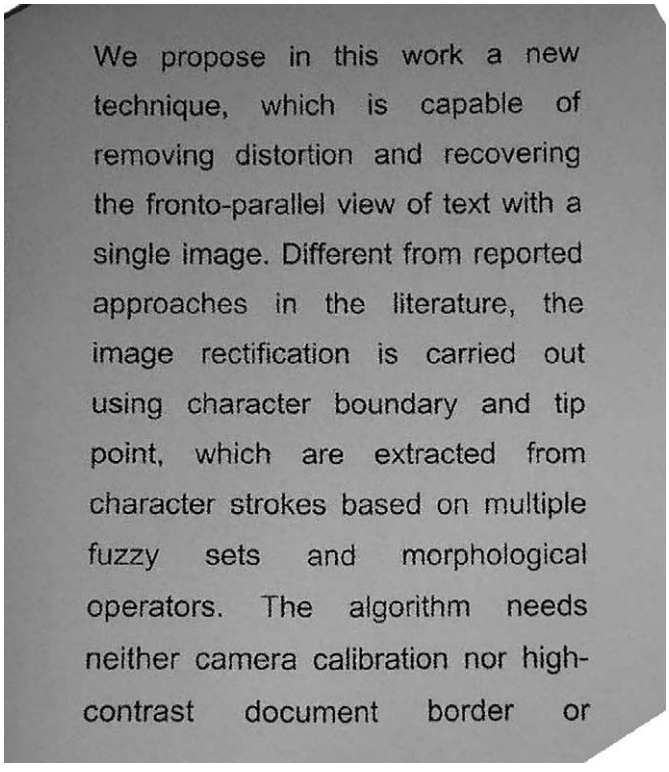

Fig. 14. Rectified document image.

2 seconds for 640×480 document images. The small variation of rectification speed is mainly due to the fact that the number and size of characters contained within the same-size document image may be different. Compared with rectification method proposed in [9], which took around 20 seconds for the rectification of the document image with 400×300 average size, the new proposed method is much faster and has the potential to be applied to real time systems.

In our experiments, we first compare the rectification results based on HDB, PF and SBTP. As the rectification result based on VPM is best compared with the one based on other PF information such as text line spacing variation [8], we only consider the rectification method based on VPM here. As Fig. 7 shows, our proposed morphological operators can extract HDB features directly. With a little adaptation, the VPM can be extracted as well. For the distorted document image shown in Fig. 15(a)–(d) show the rectification results based on HDB, VPM and SBTP features, respectively.

The proposed approach can handle document images with figures and mathematical equations as well. Fig. 16(a) shows a captured document image with figure, where the ellipse within it reflects the perspective distortion. Fig. 16(b) shows the rectified image based on SBTP features extracted from the characters within the document image. Fig. 16(c) shows the document image with mathematical equation. It can be similarly rectified based on SBTP features. Fig. 16(d) shows the rectification result.

To show the robustness and versatility of rectification method based on SBTP, we test some difficult document images. Fig. 17(a) and (c) show two document images captured by digital camera. In Fig. 17(a) there is neither HDB nor VPM information. Therefore, this document

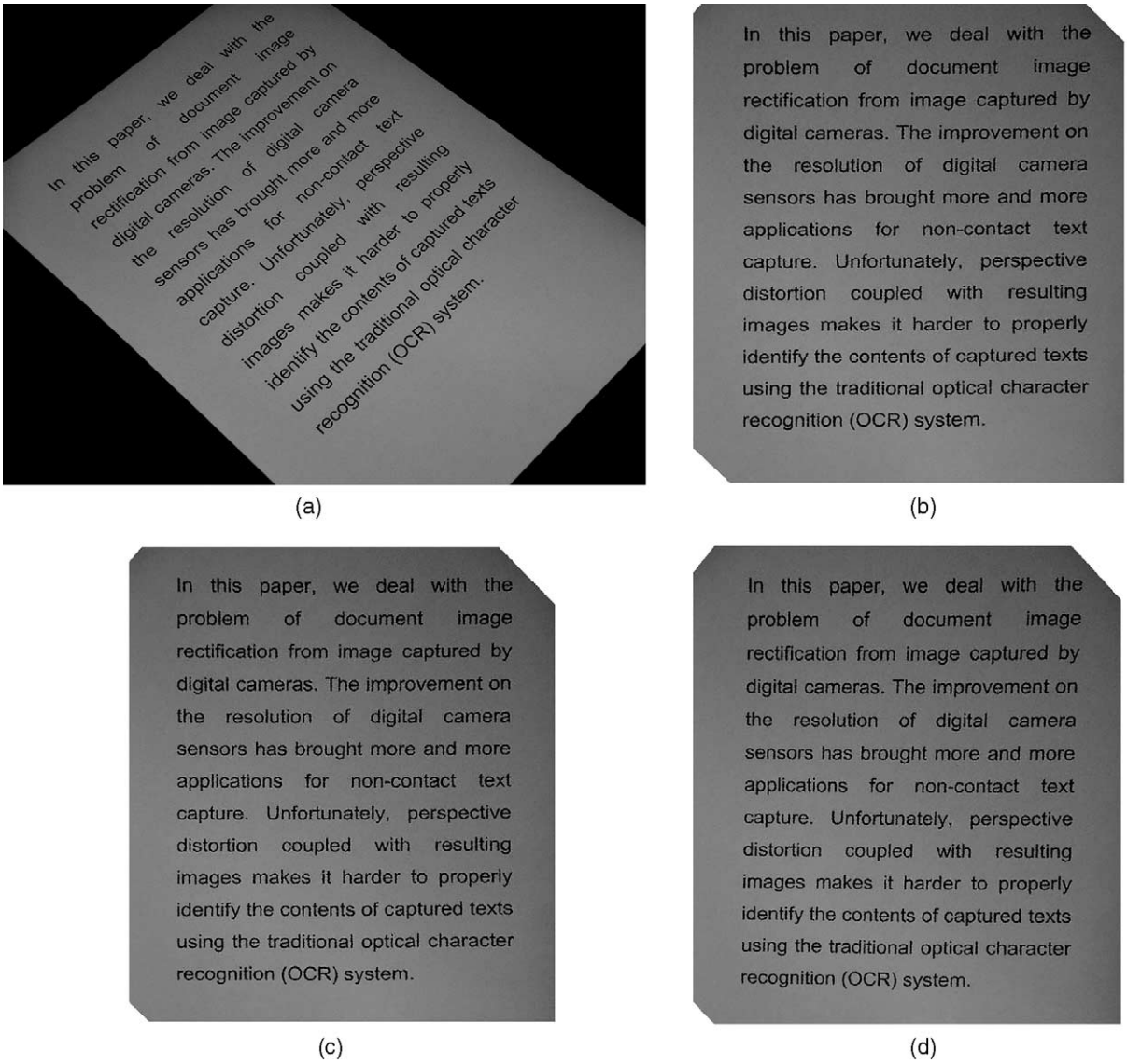

Fig. 15. Experiment results: (a) distorted document images; (b) rectified document images based on HDB; (c) rectified document image based on VPM; (d) rectified document image based on SBTP.

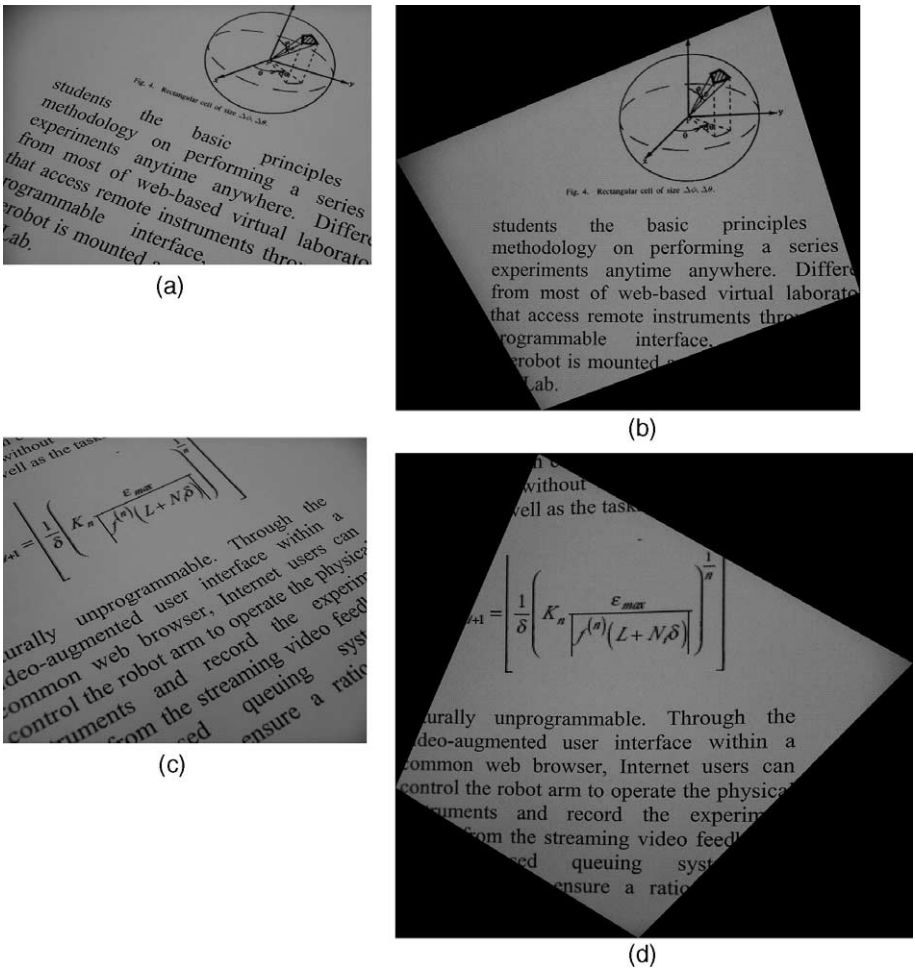

Fig. 16. Experiment results: (a), (c) distorted document images with figure and mathematical equation; (b), (d) rectified document images based on SBTP.

image cannot be rectified through the approaches proposed in [6] and [8], which requires the quadrilateral formed by HDB or VPM. However, it may be rectified by approaches proposed in [9] based on some other PF information such as text line spacing variation. Fig. 17(c) shows the enlarged document image that contains only two words. None of reported approaches in [6–9] can rectify this image since it contains neither HDB nor PF information. Fig. 17(b) and (d) show the rectification results based on extracted SBTP features.

Another important advantage of rectification based on SBTP is that it is able to rectify geometrically distorted document images where text lie on slightly curved instead of plane surface. At the present stage, the proposed technique can only handle the geometric distortion where text line is smoothly curved and the orientation of it can be approximated using the conic such as the parabola. At the same time, our method cannot flatten the captured document surface either. Under the assumption that no self-occlusion exists within the captured images, the same sets of morphological operators as used in the rectification of perspectively distorted images can be exploited to extract the SBTP features. Vertical stroke boundary and tip points can be identified and classified in the similar way. However, the horizontal curves may introduce more error and the rectification result may become poorer compared with that of planar case. Fig. 17(e) shows one document image where text lies on a curved surface and Fig. 17(f) shows the rectified image.

Since the rectification aim is to facilitate the recognition of text in images, recognition rate of rectified document image is one problem of concern. In our experiments we test the recognition rate through an OCR software Cuneiform Pro 6.0 [20]. For all the distorted document images, the recognition rates are zero. This result can be expected since traditional OCR software cannot deal with perspective distortion.

Table 1 shows the recognition rate of the rectified images. As we can see, the recognition rate is different for different images with different rectification methods. The second and third rows show the variation of recognition rate based on HDB, VPM and SBTP features. As the recognition rate based on HDB and SBTP is a bit higher than that based on VPM, we choose to use HDB feature because of its simplicity when it is available or else resort to SBTP in most of cases for rectification. The latitudinal variation of

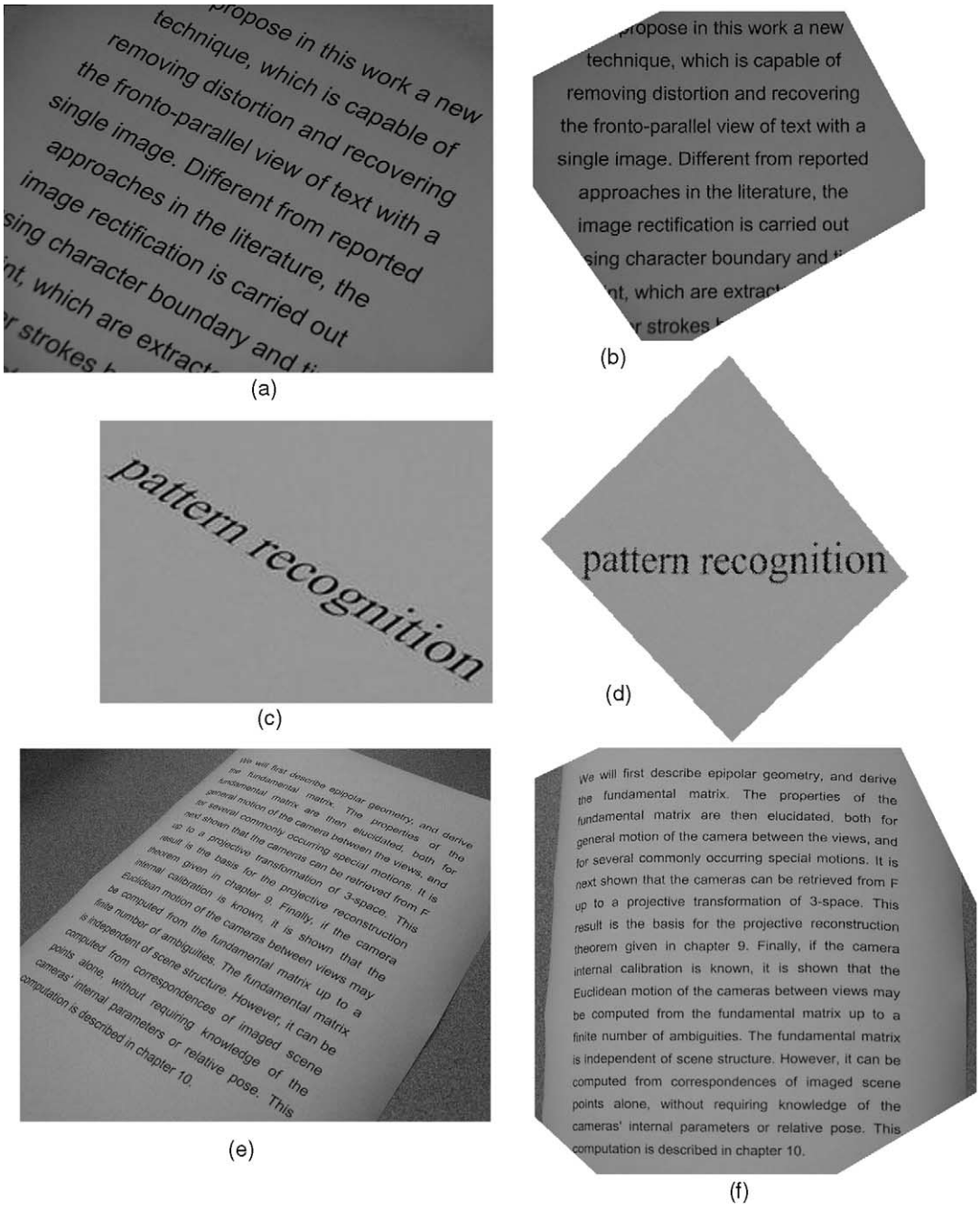

Fig. 17. Experiment results: (a), (c), (e) distorted document images; (b), (d), (f) rectified document images based on SBTP.

recognition rate with same rectification method shown in column 4 mainly results from the difference of distortion degree. We have tested a large number of perspectively distorted document images and the recognition rate of rectified images using SBTP features can averagely reach above 95%, a bit higher than that using VPM feature and lower than that using HDB feature where both VPM and HDB features are available.

Lastly, though the proposed method can deal with the text typed in different fonts such as 'Arial', 'Verdana', and 'Time New Roman', it cannot handle the document images where text is too small. Experiment results show that the rectification

Table 1
Comparison of recognition rates (RR) resulting from different rectification methods

Source image	RR for HDB	RR for VPM	RR for SBTP
Fig. 3(a)	99.37%	99.21%	99.28% (Fig. 13)
Fig. 15(a)	98.88% (Fig. 15(b))	98.59% (Fig. 15(c))	98.67% (Fig. 15(d))
Fig. 16(a)	N/A	N/A	95.32% (Fig. 16(b))
Fig. 16(c)	N/A	N/A	97.53% (Fig. 16(d))
Fig. 17(a)	N/A	N/A	100% (Fig. 17(b))
Fig. 17(c)	N/A	N/A	100% (Fig. 17(d))

performance was not apparently affected while the average size of identified vertical stroke boundaries is equal or bigger than 8 pixels, which is nearly equal to the average size of vertical stroke boundaries of size 10 'Arial' font. However, the proposed method may fails to identify some desired vertical stroke boundaries when text is becoming smaller. In addition, the proposed method may fail to rectify some illegible handwritten text where the vertical stroke boundaries cannot be properly identified either.

4. Conclusion

In this paper, a computationally efficient technique is proposed to rectify the perspectively distorted document image captured by digital camera in three-dimension space. Experimental results over a large number of distorted document images with different features show that the proposed rectification method is robust, fast, and accurate. The main advantage of the proposed technique is that it extracts the required features directly from the character strokes and accordingly, it overcomes the problems of most formerly reported methods, which are heavily dependent on some image features such as HDB and PF that do not always exist within the captured document images. At the same time, with some customized fuzzy sets and morphological operators, the proposed method executes much faster because it needs no time-consuming projection profile analysis and vanishing point searching operations required in other reported methods. Though working well on the rectification of document image of typewritten text captured by digital camera, the proposed method may fail to rectify some handwritten text because it may fail to identify the desired vertical stroke boundaries properly. With the same reason, the proposed method may fail to rectify some typewritten text with bad resolution.

With a digital camera, our proposed rectification-recognition technique may open a new channel for document digitalization and understanding. Furthermore, it may be transplanted to some other camera sensor equipped devices such as mobile phone and personal digital assistant. Some more advanced techniques that are able to handle the handwritten and geometrically distorted text will be exploited in our future work.

References

[1] H.K. Kwag, S.H. Kim, S.H. Jeong, G.S. Lee, Efficient skew estimation and correction algorithm for document images, Image and Vision Computing 20 (2002) 25–35.
[2] B. Yu, A.K. Jain, A robust and fast skew detection algorithm for generic documents, Pattern Recognition 29 (10) (1996) 1599–1629.
[3] E. Kavallieratou, N. Fakotakis, G. Kokkinakis, Skew angle estimation for printed and handwritten documents using the Wigner–Ville distribution, Image and Vision Computing 20 (2002) 813–824.
[4] L. O'Gorman, The document spectrum for page layout analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (11) (1993) 1162–1173.
[5] R.M. Haralick, Monocular vision using inverse perspective projection geometry: analytic relations, Proceedings of the IEEE Computer Vision and Pattern Recognition Conference 1989; 370–378.
[6] P. Clark, M. Mirmehdi, Recognizing text in real scenes, International Journal of Document Analysis and Recognition 4 (4) (2002) 243–257.
[7] M. Pilu, Extraction of illusory linear clues in perspectively skewed documents, Proceedings of the IEEE Computer Vision and Pattern Recognition Conference 2001; 363–368.
[8] C.R. Dance, Perspective estimation for document images, Proceedings of the SPIE Conference on Document Recognition and Retrieval IX 2002; 244–254.
[9] P. Clark, M. Mirmhedi, Rectifying perspective views of text in 3Dscenes using vanishing points, Pattern Recognition 36 (2003) 2673–2686.
[10] W.B. Irving, Applied Statistical Methods, Academic Press, New York, 1974.
[11] R. Jain, R. Kasturi, B.G. Schunck, Machine Vision, McGraw-Hill, New York, 1995.
[12] R. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, Cambridge, 2000.
[13] Y. Yang, H. Yan, An adaptive logical method for binarization of degraded document images, Pattern Recognition 33 (2000) 787–807.
[14] Y. Liu, S.N. Srihari, Document image binarization based on texture features, IEEE Transactions on Pattern Recognition and Machine Intelligence 19 (1997) 540–544.
[15] L. O'Gorman, Binarization and multithresholding of document images using connectivity, Graphical Models and Image Processing 56 (1994) 496–506.
[16] N. Otsu, A threshold selection method from grey-level histograms, IEEE Transactions on System, Man, Cybernetics SMC-8 (1978) 62–66.
[17] P. Soille, Morphological Image Analysis: Principles and Applications, second ed., Springer Verlag, Berlin, 2003.
[18] L.A. Zadeh, Calculus of Fuzzy Restrictions, Fuzzy Sets and Their Application to Cognitive and Decision Making Processes, Academic Press, New York, 1975.
[19] H.J. Zimmermann, P. Zysno, Latent connectives in human decision making, Fuzzy Sets and Systems 4 (1980) 37–51.
[20] http://www.ocr.com/.