

Character Recognition under Severe Perspective Distortion

Linlin Li, Chew Lim Tan

School of Computing, National University of Singapore

{lilinlin,tancl}@comp.nus.edu.sg

Abstract

A common problem encountered in signboard recognition is the perspective distortion of characters. In this paper, we propose a method which is able to directly recognize characters under severe perspective distortion without perspective rectification. In this method, a character is represented by a sequence of cross ratio spectra, in which the perspective effect can be modeled as an one-dimensional uneven stretching. Dynamic Time Warping algorithm is employed to estimate the pairwise similarity between spectra of the query and spectra of a fronto-parallel template. Then, it is again used to find out the pixel-level correspondence and the similarity between the query and the template. The experiment results showed that the proposed method worked well on synthetic character images and signboards in real scene under severe perspective projections.

1 Introduction

Current OCR techniques assume that the image to be recognized is a parallel projection of the original document. However, this assumption no longer holds when it comes to images captured by cameras, which often suffer from perspective distortion.

Many efforts have been made to remove the perspective distortion and rectify the distorted image into a frontal-parallel view [1] [2] [4] [5]. One important category of them is to recover the perspective projection matrix by estimating the vanishing points [1] [2] [4]. However, these methods assume that the text body has sufficient text lines and that the layout is highly formatted. Another category, which is able to handle individual text lines, approximates the perspective transformation by an affine transformation [5]. The approximation holds when the distance between the object and the camera is much greater than the size of the object. There were also a few works to recognize individual charac-

ters under perspective distortion directly using structural invariants, without any rectification [3] [6]. However, the performance of detection of those invariants, such as ascender and descender, highly depends on the length of text lines, hence this method may fail when a text line is too short. In order to deal with the perspective distortion, methods mentioned above put constraints either on the perspective angle (weak perspective) or the contextual information (text line). In this paper, a method to recognize characters under severe perspective distortion making no assumption about the perspective angle or contextual information, will be presented.

2 Approach

Cross Ratio is a fundamental invariant for perspective transformation. The cross ratio of four collinear points (P_1, P_2, P_3, P_4) displaying in order is defined as:

$$\text{cross_ratio}(P_1, P_2, P_3, P_4) = \frac{P_1P_3}{P_2P_3} / \frac{P_1P_4}{P_2P_4} \quad (1)$$

where P_iP_j denotes the distance between P_i and P_j . $\text{cross_ratio}(P_1, P_2, P_3, P_4)$ remains unchanged under any perspective transformation. In the rest of this paper, the notation Q refers to the query character as well as the pixel sequence along the convex hull of the character, which is denoted by $\{Q_1, \dots, Q_n\}$. Similarly, T refers to a template character and its pixel sequence on the convex hull, denoted by $\{T_1, \dots, T_m\}$. The recognition process has two steps: **1.** Compare the pairwise similarity of spectra of T and spectra of Q . **2.** Find the pixel-level correspondence and estimate the similarity between T and Q .

2.1 Cross Ratio Spectrum

A cross ratio spectrum is a sequence of cross ratio values. Figure 1 shows a character 'H' under fronto-parallel view (P) and perspective view (P'). Suppose pixels $P_1 \in P$ and $P_k \in P$ have mapping pixels

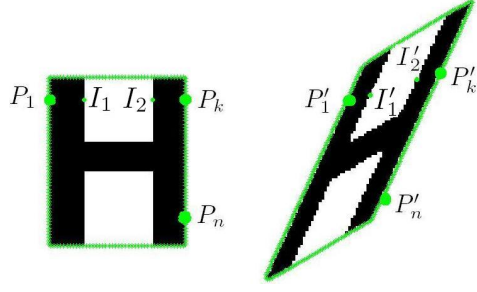


Figure 1. Character 'H' in the fronto-parallel view and a perspective view.

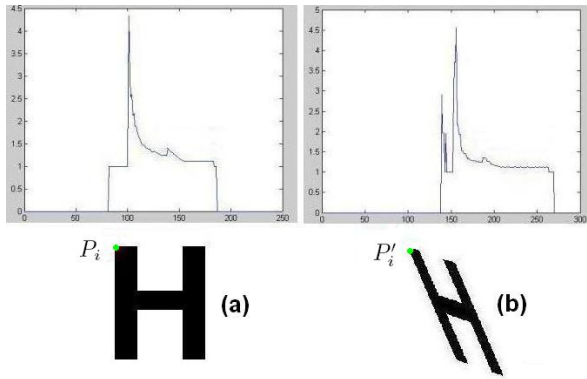


Figure 2. Cross Ratio Spectra of two mapping pixels P_i and P'_i .

$P'_1 \in P'$ and $P'_k \in P'$, respectively. Then I_1 and I_2 (intersections of the strokes and line P_1P_k) have mapping pixels I'_1 and I'_2 (intersections of the strokes and the line $P'_1P'_k$). Consequently, the following equation holds:

$$\text{cross_ratio}(P_1, I_1, I_2, P_k) = \text{cross_ratio}(P'_1, I'_1, I'_2, P'_k) \quad (2)$$

Suppose the pixel sequence of the convex hull of P is $\{P_1, P_2, \dots, P_n\}$, where P_2 is the anti-clock-wise neighbor pixel of P_1 , and P_3 is the anti-clock-wise neighbor pixel of P_2 , and etc. Cross ratio values are calculated from each pair of (P_1, P_j) and intersections between them. When there are more than two intersections between both pixels, such as P_1 and P_n in figure 1, only the first two intersections are used. For simplicity, we will rewrite the cross ratio notation and leave out the intersections as follows:

$$CR(P_1, P_k) = \text{cross_ratio}(P_1, I_1, I_2, P_k) \quad (3)$$

If the number of intersections is 0 or 1, when no cross ratio value can be computed, the pseudo-cross ratio value is assigned as -1 or 0 respectively. The cross ratio value of characters ranges from 1 to ∞ . This assignment is to guarantee that pseudo-cross ratio is distinct from a real one. A Cross Ratio Spectrum (CRS) of a pixel P_i is defined as: $CRS(P_i) = \{CR(P_i, P_{i+1}), \dots, CR(P_i, P_n), CR(P_i, P_1), \dots, CR(P_i, P_{i-1})\}$.

2.2 Modeling The Perspective Effect

Although a cross ratio value remains constant under perspective projection, the spectrum of a pixel does change. Because under a perspective projection, some parts of a character expand, while some parts shrink. This leads to pixel increasing or decreasing on certain parts of the convex hull. As a result, some segments of the spectrum curve are elongated, while some are shortened. An example is shown in figure 2. The cross ratio spectra of two mapping pixels P_i and P'_i are shown in figure 2(a) and (b) respectively, where x-axis is the pixel index and y-axis is the cross ratio value (for a better view purpose, all pseudo-cross ratios are set as 0). Visually, spectrum (b) is very similar to spectrum (a) except for a few noise, but it has certain fluctuation in the x-axis. In our method, spectrum (b) is modeled as an uneven stretching version of curve(a). Hence, we use Dynamic Time Warping (DTW) algorithm to compare the similarity between spectrum (a) and (b). It is worth noting that, the noise is caused by false pixel quantization in corners. When the distance between two intersections is long enough, increasing or reducing one-pixel length introduced by the quantization will not affect the cross ratio value; otherwise the cross ratio may change a lot.

2.3 Comparing Cross Ratio Spectra

$CRS(Q_i) = \{CR(Q_i, Q_{i+1}), \dots, CR(Q_i, Q_{i-1})\}$ is rewritten as $CRS(Q_i) = \{q_u, u = 1 : \mu\}$ for simplicity. Similarly, $CRS(T_j) = \{t_v, v = 1 : \nu\}$. The comparison between $CRS(Q_i)$ and $CRS(T_j)$ is formulated as:

$$DTW(u, v) = \min \begin{cases} DTW(u-1, v-1) + c(u, v) \\ DTW(u-1, v) + c(u, v) \\ DTW(u, v-1) + c(u, v) \end{cases} \quad (4)$$

where $c(u, v) = \text{abs}(q_u - t_v)/(q_u + t_v)$ is the cost function. The similarity score is given by $DTW(\mu, \nu)$ over the length of the warping path.

	Q_1	...	Q_m	Q_1	...	Q_m
T_n						
...						
T_1						

Table 1. The global similarity table.

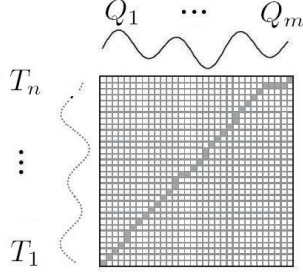


Figure 3. Global similarity table searching using DTW.

2.4 Comparing Query and Template

The DTW comparisons are conducted between each pair of Q_i and T_j , and a $2 \times m \times n$ global similarity table is constructed as table 1 (although $m \times n$ comparisons are conducted), cell (i, j) denotes the similarity of corresponding pixels. After the global similarity table is filled, the pixel-alignment path is found in the table.

Each time, a DTW is applied to a sub-table comprised of column $\{\bar{h}, \bar{h} + 1, \dots, \bar{h} + m - 1\}$ of the global table, to align T_1 with $Q_{\bar{h}}$ and T_n with $Q_{\bar{h}+m-1}$ as the initial condition. The comparison is formulated as follows:

$$DTW(i, j) = \min \begin{cases} DTW(i-1, j-1) + c(i, j) \\ DTW(i-1, j) + c(i, j) \\ DTW(i, j-1) + c(i, j) \end{cases} \quad (5)$$

where $c(i, j) = \text{global_similarity_table}(\bar{h} + i - 1, j)$, $i = 1 : m$, and $j = 1 : n$. An example is shown in figure 3 when $\bar{h} = 1$. The path labeled in gray color is the desirable path, and the similarity score between Q and T is given by $DTW(m, n)$ over the length of the path. Among m possible scores, the smallest one gives the similarity score that is desirable. Meanwhile, the pixel level correspondence between Q and T is given by the warping path.

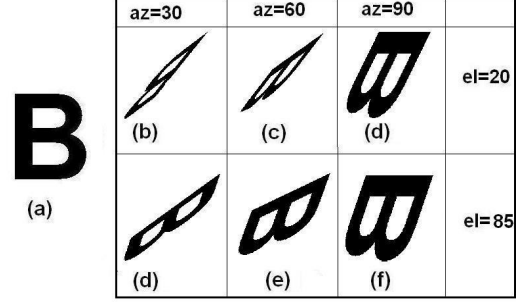


Figure 4. Generated character images.

el=	20°	30°	40°	55°	70°	85°
az=30°	91.94	93.55	93.55	100	100	100
az=60°	93.55	96.77	100	100	100	100
az=90°	100	100	100	100	100	100

Table 2. Recognition results of synthetic images.

3 Experiment Results and Discussion

A template set was trained on synthetic fronto parallel images of 26 uppercase English characters, 26 lowercase characters and 10 digits in Arial font and bold style. In the experiments, testing characters were compared with the template set. The template character which had the highest similarity with the query gave the identity of the query.

3.1 Synthetic Image Testing

In the first experiment, 18 testing datasets, each of which comprised of 26 uppercase characters, 26 lowercase characters and 10 digits, were generated by matlab using various perspective parameters. The characters are of Arial font and bold style. The perspective images were generated by setting the target point at the center of a 100×100 pixel bounding box around each character, and setting the perspective angle as 25° , while changing the azimuth and elevation angle gradually. The smallest elevation angle was set as 20° , because further reduction of the elevation angle led to characters which are difficult for human to recognize. Examples of the character 'B' under different perspective distortions are shown in figure 4.

Table 2 shows the percentage of correctly recognized characters under different perspective distortions. The result gets 100% accuracy when the azimuth angle is

90°, while some degradation happens when the azimuth angle is 60° or 30°. The result shows that the proposed method works well under severe perspective distortion. On the contrary, two OCR softwares used in the experiment, namely Tesseract OCR and OminiPage Pro 14.0, almost were not able to recognize these characters. For the proposed method, errors happened within character groups {DO} and {CU}. Because these characters have very similar spectrum sequences. Noise (abnormal peaks caused by false pixel quantization) in the spectrum will have a significant impact on the recognition. The problem may be addressed by selecting a better cost function other than square difference which tends to magnify the effect of noise. However, the proposed method still showed strength to differentiate characters with similar structures, like {OO}, {ZN}, {L7}, and {UV}. Even in fronto-parallel view, these characters could be easily mistaken for each other in recognition when their orientations are unknown.

3.2 Real Scene Image Testing

The purpose of the second experiment is to investigate how the proposed method works when real scene noise appears, introduced by the camera lens or the binarization process. 20 photos of different signboards taken from different angles were tested. The testing photos were 1200 × 1600 pixels with similar but not exact fonts as Arial. In order to show the recognition performance independently and avoid errors caused by character detection, images were binarized by a global thresholding method and then character boxes of English characters and digits were manually extracted before recognition. We got 294 characters in total. Examples of the testing photos are shown in figure 5.

The recognition accuracy is 100% for our method, 30.61% for Tesseract OCR, and 61.2% for Ominipage Pro. The performance is better than that in the first experiment. The reason for higher recognition accuracy is that, due to physical constraints during taking photos, the perspective distortion appearing in the real scene images was not as severe as those in the synthetic data. Note that these signboards have fonts similar to the training font, but not always the same. Thus the performance suggests that the proposed method is tolerant to different fonts to a certain degree. In the third photo of figure 5, the word “RESERVED” is elongated in the vertical direction in order to fit the word into the signboard, which is very common in signboard making. In other words, the elongating effect in the vertical direction exists even in the fronto-parallel view. A few testing images are under bad illumination, but the proposed method shows a certain robustness to imperfectly



Figure 5. Signboards in real scene.



Figure 6. A photo with bad illumination.

segmented characters from these images. For example, the string “STOREY” shown in figure 6 was correctly recognized in the experiment.

4 Conclusion

This paper proposes a new character recognition method based on cross ratio spectrum. This technique is capable of directly recognizing characters with an elevation angle as small as 20°. Because of DTW, the method currently suffers from long execution time. However, further optimizing the comparison process by DTW indexing is possible. Although only English characters and digits were tested in the experiment, we believe that this method can be extended to all Latin-based languages and some simple symbols. Further examination of the discriminating power of the method on a larger character set will be done in future. In addition, issues of fonts and typefaces will also be addressed in a more detailed way too.

Acknowledgment: This research is supported in part by IDM R&D grant R252-000-325-279.

References

- [1] P. Clark and M. Mirmehdi. Recognizing text in real scenes. *IJDAR*, 4(4):243–257, 2004.
- [2] S. Lu, B. Chen, and C. Ko. Perspective rectification of document images using fuzzy set and morphological operations. *IVC*, 23(5):541–553, 2005.
- [3] S. Lu and C. L. Tan. Camera text recognition based on perspective invariants. *ICPR*, 2:1042–1045, 2006.
- [4] M. Pilu. Extraction of illusory linear clues in perspective skewed documents. *CVPR*, 1(4):363–368, 2001.
- [5] G. Myers, R. Bolles, Q. T. Luong, and J. Herson. Recognition of text in 3-d scenes. *SDIUT*, pages 85–100, 2001.
- [6] T. Yamaguchi, M. Maruyama, H. Miyao, and Y. Nakano. Digit recognition in a natural scene with skew and slant normalization. *IJDAR*, 7(2-3):168–177, 2005.