# Recognition of Character Strings in Low-quality Images Using Character and Inter-character Space Patterns

Hiroyuki Ishida, Tomokazu Takahashi, Ichiro Ide and Hiroshi Murase
Graduate School of Information Science, Nagoya University
Furo-cho, Chikusa-ku, Nagoya, Aichi, 464-8601, Japan
{hishi, ttakahashi, ide, murase}@murase.m.is.nagoya-u.ac.jp

## Abstract

*A method for character-string recognition is presented. The proposed method jointly evaluates features obtained from individual characters and inter-character spaces. Combining these features resolves ambiguity in segmentation and classification of low-quality character-string images. To evaluate the inter-character features, an inter-character orthogonal subspace is constructed for each permutation of two characters. Experimental results exhibited the usefulness of the inter-character features.*

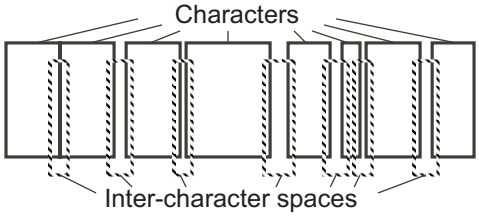


**Figure 1. Two features available in character-string recognition**

## 1. Introduction

Text recognition technologies using portable digital cameras have gained attention in recent years in proportion to the diffusion of portable digital imaging devices [1]. A challenging problem in camera-based text recognition is blurring and reduction of image resolution. Characters in such low-quality images often touch adjacent characters. For accurate recognition of character strings, they should be segmented properly. As has been discussed in many studies [2, 3], neither recognition nor segmentation of low-quality character-string images can be done independently. One solution is recognition-based segmentation [4], in which segmentation ends when the recognition result is obtained. Nevertheless, even in many recognition-based segmentation methods, the segmentation result is considered only as a by-product of recognition; the boundary of adjacent characters tends to be positioned ambiguously. A sophisticated method developed by Sun et al. [5] combines several features for segmentation and recognition. This paper focuses on inter-character spaces as shown in Fig. 1. Their features have yet to be evaluated positively and effectively.

In the method presented here, individual characters and inter-character spaces are simultaneously recognized. The latter is done by discerning between which characters spaces exists. Similarly to conventional methods, the proposed method classifies given images based on similarity, but inclusive of similarities from inter-character spaces.

This paper is organized as follows. Section 2 describes the use of features in inter-character spaces. Section 3 describes character-string recognition based on the inter-character features. Results are presented in Section 4.

## 2. Features in inter-character spaces

Inter-character space is the region from the rightmost column of a left-hand character to the leftmost column of a right-hand character. Provided that the left-hand character is $l$, and the right-hand character is $r$, the following characteristics should be noted:

**(i)** The left half of the space is similar to $l$.

**(ii)** The right half of the space is similar to $r$.

However, evaluating only these characteristics can yield too many candidates for $l$ and $r$, which leads to false segmentation. Therefore the following characteristic is aditionally velified.

**(iii)** The features in the inter-character space change remarkably from left to right.

Characteristic (iii) is important also for clearly identifying the boundaries of the adjacent characters. Our method measures them by an approach similar to the orthogonal subspace method [6, 7], in which eigenvectors of each category are orthogonal. In this case, an inter-character orthogonal subspace is constructed for two categories of adjacent characters. Characteristics (i)–(iii) are measured by the area of the triangle made with two feature vectors projected onto the inter-character orthogonal subspace. Thus both length and span of the projected vectors are evaluated.

### 2.1 Construction of projection matrix

A projection matrix to an inter-character orthogonal subspace is calculated beforehand. First of all, two feature vectors corresponding to the leftmost and the rightmost columns of the characters are needed. They are averaged from multiple training images of each category $c$ and denoted as $\boldsymbol{\phi}^{(c|}$ (leftmost column) and $\boldsymbol{\phi}^{|c)}$ (rightmost column). An inter-character orthogonal subspace is constructed for each permutation of the two categories. The process is described below.

Let $\mathcal{I}^{|l)(r|}$ be an inter-character orthogonal subspace between a left-hand character $l$ and a right-hand character $r$. Using $\boldsymbol{\phi}^{|l)}$ corresponding to the rightmost column of $l$ and $\boldsymbol{\phi}^{(r|}$ corresponding to the leftmost column of $r$, a correlation matrix $\boldsymbol{P}$ is calculated by

$$\boldsymbol{P} = \frac{1}{2}\left(\boldsymbol{\phi}^{|l)}\boldsymbol{\phi}^{|l)\top} + \boldsymbol{\phi}^{(r|}\boldsymbol{\phi}^{(r|\top}\right). \tag{1}$$

Two eigenvalues and corresponding eigenvectors are derived from this matrix $\boldsymbol{P}$ and denoted as $(\lambda_1, \lambda_2)$ and $(\boldsymbol{e}_1, \boldsymbol{e}_2)$, respectively. A projection matrix $\boldsymbol{W}^{|l)(r|}$ onto $\mathcal{I}^{|l)(r|}$ is obtained by

$$\boldsymbol{W}^{|l)(r|} = \frac{1}{\sqrt{2}}\boldsymbol{\Lambda}^{-1/2}\boldsymbol{B}^{\top} \tag{2}$$

with

$$\boldsymbol{\Lambda}^{-1/2} = \mathbf{diag}\left(\lambda_1^{-1/2}, \lambda_2^{-1/2}\right)$$
$$\boldsymbol{B} = [\boldsymbol{e}_1 \;\; \boldsymbol{e}_2].$$

Upon the projection by this matrix, it follows that

$$\left|\boldsymbol{W}^{|l)(r|}\boldsymbol{\phi}^{|l)}\right| = \left|\boldsymbol{W}^{|l)(r|}\boldsymbol{\phi}^{(r|}\right| = 1$$
$$\boldsymbol{W}^{|l)(r|}\boldsymbol{\phi}^{|l)} \perp \boldsymbol{W}^{|l)(r|}\boldsymbol{\phi}^{(r|}.$$

This means that areas on $\mathcal{I}^{|l)(r|}$ are normalized. Furthermore, we demand that a determinant $\det\left|\boldsymbol{W}^{|l)(r|}\boldsymbol{\phi}^{|l)} \;\; \boldsymbol{W}^{|l)(r|}\boldsymbol{\phi}^{(r|}\right|$ shall be positive. If not, $\boldsymbol{W}^{|l)(r|}$ needs to be reconstructed by changing the sign of $\boldsymbol{e}_2$. This operation unifies the signs of the determinant. The process of constructing $\mathcal{I}^{|l)(r|}$ described above is illustrated in Fig. 2. Some examples of eigenvectors are illustrated in Fig. 3.

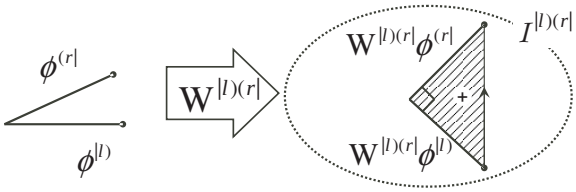


**Figure 2. Construction of inter-character orthogonal subspace $\mathcal{I}^{|l)(r|}$ from training feature vectors**

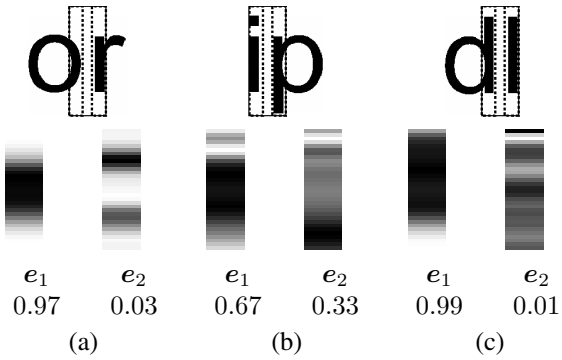


**Figure 3. Examples of eigenvectors of inter-character spaces. Values shown at bottom are eigenvalues.**

### 2.2 Recognition of inter-character space

Given an inter-character space from the $n$-th column to the $m$-th column, the similarity of the space to $\mathcal{I}^{|l)(r|}$ is calculated. Let $\boldsymbol{y}_i$ $(n \le i \le m)$ be vectors, each of which consists of pixel values in the $i$-th column. Also, let them be normalized so that the mean is 0 and the norm is 1. They are projected onto $\mathcal{I}^{|l)(r|}$, and thereby form triangles with sides $\boldsymbol{W}^{|l)(r|}\boldsymbol{y}_i$. The similarity to $\mathcal{I}^{|l)(r|}$ is defined as the sum of the area of these triangles. Accordingly,

$$s^{|l)(r|}_{(n,m)} = \frac{1}{2}\sum_{i=n}^{m-1}\det\left|\boldsymbol{W}^{|l)(r|}\boldsymbol{y}_i \;\; \boldsymbol{W}^{|l)(r|}\boldsymbol{y}_{i+1}\right|. \tag{3}$$

Figure 4 illustrates the process of the similarity calculation. In this example, a region composed of three columns is compared to an inter-character orthogonal subspace between "o" and "r".

For some combinations of very similar $\boldsymbol{\phi}^{|l)}$ and $\boldsymbol{\phi}^{(r|}$, however, the characteristics described above cannot be measured. In such case, the resulting $\boldsymbol{P}$ in Eq.(1) does not possess a valid second eigenvector. This case is found in example (c) of Fig. 3. For such a combination of $l$ and $r$,

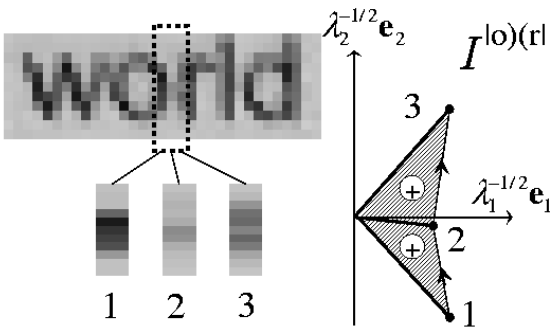


**Figure 4. Calculation of similarity. Similarity to inter-character orthogonal subspace is given as sum of areas.**

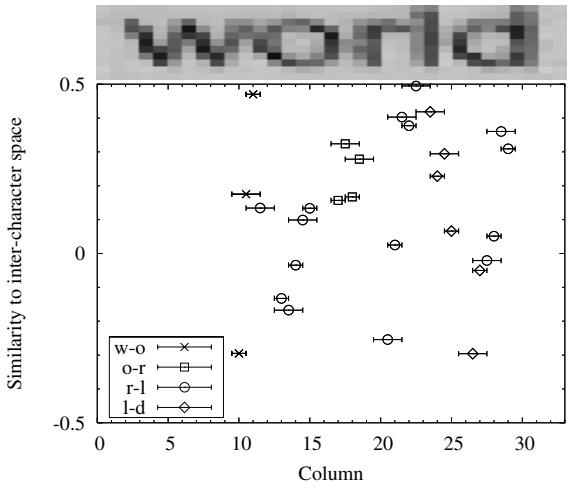


**Figure 5. Similarities calculated for various inter-character spaces. Bars corresponding to each inter-character space are plotted.**

similarity is evaluated by

$$s^{|l)(r|}_{(n,m)} = 0 \qquad (\lambda_2 < \epsilon), \tag{4}$$

with a small $\epsilon$. The purpose of this strategy is to avoid over-segmentation. If a positive similarity is given to such a combination, for example, character image "l" composed of two columns is likely to be classified to "ll". The value of $\epsilon$ is determined from measurements of the eigenvalues of such inter-character orthogonal subspaces.

## 3 Recognition of character-string image

Recognition is based on the candidate character lattice method [3, 8]. Whereas the hidden Markov model (HMM) or its extensions have been widely used for handwritten text, the lattice method is simpler and suitable for printed text even in low-resolution. The lattice method initially recognizes individual characters, and thereby a hypothesis graph with the lists of the candidate characters is constructed as illustrated in Fig. 6. Character-string recognition is performed by searching for the optimal path.

### 3.1 Individual character recognition

Characters are recognized by the subspace method [6]. The effectiveness of the subspace method for low-resolution characters is reported by Yanadume et al. in [9].

For each category $c$, a subspace $\mathcal{S}^{(c)}$ is calculated beforehand from training data by Principal Component Analysis. In a given character-string image, a region from the $m$-th column to the $n$-th column is denoted as $\boldsymbol{z}_{(m,n)}$. Let it be vectorized and normalized so that the mean is 0 and the norm is 1. Similarity to $\mathcal{S}^{(c)}$ is defined as a sum of squared inner products to eigenvectors $\boldsymbol{e}_r^{(c)} \in \mathcal{S}^{(c)}$ by

$$s^{(c)}_{(m,n)} = \sum_{r=1}^{R} \left( {\boldsymbol{e}_r^{(c)}}^{\top} \boldsymbol{z}_{(m,n)} \right)^2, \tag{5}$$

where $R$ is the number of eigenvectors.

### 3.2 Evaluation of similarities

Once a hypothesis graph is constructed, the character-string recognition is simplified to searching for an optimal path in the hypothesis graph. The recognition result is uniquely determined as a list of candidate characters.

Let $c_j$ be the $j$-th character in a path ($1 \le j \le J$), $m_j$ be the leftmost column of $c_j$, and $n_j$ be the rightmost column of $c_j$, where

$$m_1 < n_1 < m_2 < n_2 < \cdots < m_J < n_J.$$

The sum of the similarities to individual characters is defined as

$$S_1 = \sum_{j=1}^{J} (n_j - m_j + 1) s^{(c_j)}_{(m_j,n_j)}. \tag{6}$$

Meanwhile, the sum of the similarities to inter-character spaces is defined as

$$S_2 = (n_J - m_1 + 1) \sum_{j=1}^{J-1} \left[ s^{|c_j)(c_{j+1}|}_{(n_j,m_{j+1})} - 1 \right]. \tag{7}$$

Actually, this $S_2$ acts as a penalty, since it is negative. A joint similarity $S$ is defined as the weighted sum of these

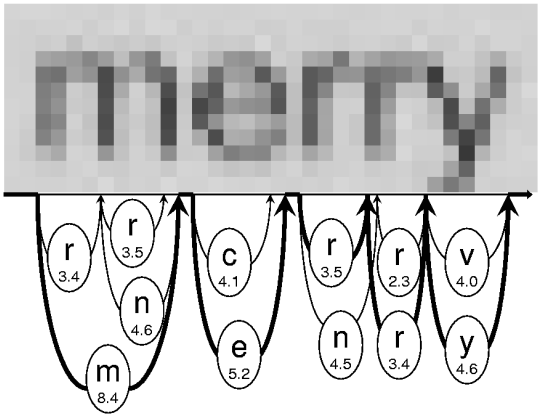


**Figure 6. Conventional hypothesis graph constructed for character-string image. Candidate characters are shown with their similarities. Path shown with bold lines maximizes the sum of the similarities.**

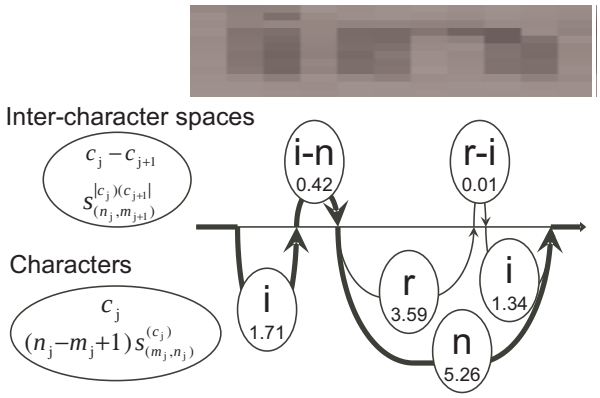


**Figure 7. Example of hypothesis graph using inter-character features.**

similarities. Using weight $k$, $S$ is calculated by

$$S = S_1 + kS_2. \tag{8}$$

The recognition result $(c_1, c_2, ..., c_J)$ is obtained from the optimal path maximizing this $S$. An appropriate $k$ for recognition needs to be determined experimentally.

## 4 Experiment

In this section, the effectiveness of the proposed method is evaluated. A digital camera (Panasonic DMC-FX9) was used to take character-string images. 233 words printed on paper were captured 20 times, and in all 4,660 character-string images were used as test data. The number of categories was 62 (A–Z, a–z, 1–9: Ariel font). The average size of the character strings in the images was $33.0 \times 12.0$ pixels. In the process of extracting the character-string images, their height was initially estimated from the whole document image, and the areas to be segmented were then determined such that each of the contained character string was located at the center of the area.

In the training step, all training images were synthesized from original templates of character images (Ariel font) by a generative learning method [10]. To cope with segmention errors, variously shifted images were used for the training. By shifting horizontally and vertically, 625 training images were generated for each category. They were then normalized to images of $32 \times 32$ pixels. Parameter $\epsilon$ in Eq. (4) was set to $0.02$. The number of eigenvectors $R$ in Eq. (5) was set to $5$.

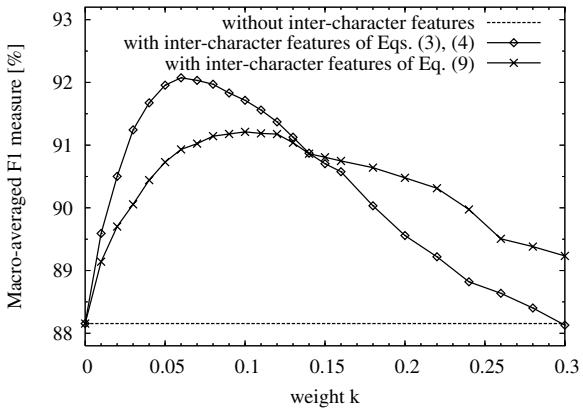


**Figure 8. Recognition results**

Recognition results for various $k$ in Eq.(8) are presented in Fig. 8. A macro-averaged $F_1$ measure [11] was used for the evaluation, where $F_1$ is given for each test character-string by the formula $F_1 = 2pr/(p+r)$ with precision rate $p$ and recall rate $r$. Letting $\mathcal{C}$ and $\mathcal{R}$ denote sets of characters in the correct string and in the recognized string, respectively, it follows that $p = |\mathcal{C} \cap \mathcal{R}|/|\mathcal{R}|$ and $r = |\mathcal{C} \cap \mathcal{R}|/|\mathcal{C}|$.

According to the results, the recognition accuracy increased while $k$ was small ($k < 0.06$), indicating that the features obtained from the inter-character spaces are capable of resolving the ambiguity in the classification of low-quality character-string images. However, the recognition accuracy decreased once $k > 0.06$. This result showed that the inter-character features were relatively poor in stability. Figure 9 shows some examples of recognition results. Setting weight $k$ higher than zero eliminated some segmentation errors but simultaneously yielded new errors.

| | | | | |
|---|---|---|---|---|
| Correct words | world | rooms | on | but |
| Captured images |  |  |  |  |
| $k = 0$ | worlidl | iroorns | on | 1but |
| $k = 0.06$ | world | rooms | on | but |
| $k = 0.30$ | world | rooms | oln | but |
| $k = 0.10$ [Eq.(9)] | world | rooms | on | lblut |

**Figure 9. Examples of test data and their recognition results. Results at the bottom are obtained using Eq. (9).**

Other results in Fig. 8 were obtained from a simple approach, which evaluates inner products to the training vectors $\boldsymbol{\phi}^{|l)}$ and $\boldsymbol{\phi}^{(r|}$ in Section 2.1 as the similarity. This approach covers only characteristics (i) and (ii) in Section 2. In this case, the similarity in Eq.(3) was calculated by

$$s^{|l)(r|}_{(n,m)} = \frac{1}{2}\left(\boldsymbol{\phi}^{|l)\top}\boldsymbol{y}_n\right)\left(\boldsymbol{\phi}^{(r|\top}\boldsymbol{y}_m\right). \qquad (9)$$

However this approach did not achieve sufficient performance. Compared with the proposed method, the approach by Eq.(9) fails to evaluate the difference between $\boldsymbol{y}_n$ and $\boldsymbol{y}_m$. This result showed the advantage of the proposed method which evaluates the area of the inter-character orthogonal subspace.

## 5 Conclusion

In this paper, a recognition method for low-quality character-string images is proposed. In order to improve the accuracy of recognition and segmentation, features both in individual characters and inter-character spaces are used jointly. The usefulness of the features in the inter-character spaces was experimentally shown. A better way to combine these features and optimal values of the parameter $k$ for various degrees of image degradation should be discussed further in future work. Extending the dimensions of inter-character orthogonal subspaces is another interesting and important consideration.

## Acknowledgement

Parts of this research were supported by the Grants-In-Aid for Scientific Research (16300054 and 17650050) and the 21st century COE program from the Ministry of Education, Culture, Sports, Science and Technology. This work is implemented based on the MIST library (*http://mist.suenaga.m.is.nagoya-u.ac.jp/*).

## References

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

[2] C. Mancas-Thillou, M. Mancas, and B. Gosselin, "Camera-based degraded character segmentation into individual components," Proc. 8th Int. Conf. on Document Analysis and Recognition, pp.755–759, Seoul, Korea, August 2005.

[3] S. Wachenfeld, H. Klein, and X. Jiang, "Recognition of screen-rendered text," Proc. 18th Int. Conf. on Pattern Recognition, pp.1086–1089, Hong Kong, China, August 2006.

[4] R. Casey and E. Lecolinet, "A survey of methods and strategies in character segmentation," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.18, no.7, pp.690–706, July 1996.

[5] J. Sun, Y. Hotta, K. Fujimoto, Y. Katsuyama, and S. Naoi, "Grayscale feature combination in recognition based segmentation for degraded text string recognition," Proc. 1st Int. Workshop on Camera-Based Document Analysis and Recognition, pp.39–44, Seoul, Korea, August 2005.

[6] E. Oja, "Subspace methods of pattern recognition," Research Studies, Hertfordshire, UK, 1983.

[7] T. Kawahara, M. Nishiyama, and O. Yamaguchi, "Face recognition by orthogonal mutual subspace method (In Japanese)," IPS Japan SIG Technical Report, CVIM–151, pp.17–24, November 2005.

[8] H. Murase, T. Wakahara, and M. Umeda, "Online writing-box free character string recognition by candidate character lattice method (in Japanese)," IEICE Trans., vol.J-68-D, no.4, pp.765–772, April 1985.

[9] S. Yanadume, Y. Mekada, I. Ide, and H. Murase, "Recognition of very low-resolution characters from motion images," Proc. PCM2004, Lecture Notes in Computer Science, Springer-Verlag, vol.3331, pp.247–254, December 2004.

[10] H. Ishida, S. Yanadume, T. Takahashi, I. Ide, Y. Mekada, and H. Murase, "Recognition of low-resolution characters by a generative learning method," Proc. 1st Int. Workshop on Camera-Based Document Analysis and Recognition, pp.45–51, Seoul, Korea, August 2005.

[11] Y. Yang, "An evaluation of statistical approaches to text categorization," Journal of Information Retrieval, vol.1, no.1–2, pp.69–90, April 1999.