Matching Topological Structures for Handwritten Character Recognition

Daw-Ran Liou, Yang-En Chen, and Cheng-Yuan Liou(✉)

Department of Computer Science and Information Engineering,
National Taiwan University, Taipei, Taiwan
cyliou@csie.ntu.edu.tw

Abstract. This work presents a locking and deforming process to accomplish the recognition. The best matched features of the template are locked to their target features of the unknown pattern. The whole template is then deformed and calibrated according to these features. Improved similarity score can be obtained from the deformed template. This work illustrates this process and its operations. This process indirectly overcomes difficult distortion problems.

Keywords: Pattern recognition · Image recognition · Image classification · Image restoration · Robot vision · Handwritten character recognition

1 Introduction

The spatial topology distance method (STD) [9, 10] used the self-organizing algorithm (SOM) [4] to evolve the template toward its target, unknown handwritten character. This evolution can preserve the topological structure of the template and resolve serious distortions in the character. The computation cost of the evolution is high. To reduce the number of iterations in the evolution, geometrical constraints among features are manually constructed to reduce the searching operation in STD. These constraints can be converted into a set of rules [6, 7], Fig. 1, and saved in a Hopfield network [7, 11]. The number of such rules is countless. It is tedious to list the enormous number of rules manually. The compatibility of the connection rules for a whole template can be loaded in the Hopfield network [7, 11]. It can be obtained from the converged network [12, 13]. So far, there is no efficient procedure for the construction of such rules. This work devises an alternative way to skip those rules in the recognition. The same features used to represent the character and the template [11] will be used in this work. We briefly review the feature and related works in this section.

The unknown character and all its templates are prepared in advance by their feature collections that resemble the receptive fields [1–3] of visual system. All collections are loaded with flexible and variable features along their skeletons. The global operations, shift, rotation, and scaling, are applied to deform the template according to certain highly matched, locked, features. After these operations, the locked features in the deformed template will align with their corresponding features in the unknown character perfectly. The improved whole similarity score can be obtained for all features of the deformed template. This kind deforming and locking process can be

N. T. Nguyen et al. (Eds.): ACIIDS 2019, LNAI 11431, pp. 565–575, 2019.
https://doi.org/10.1007/978-3-030-14799-0_49

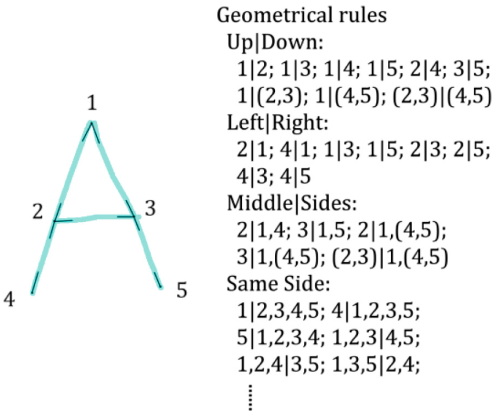

Fig. 1. Geometrical rules for the five features of character “A”.

operated iteratively. The recognition is accomplished for the template which received the highest whole score.

The templates and unknown character are both represented by collections of bended-ellipse features [10, 11], Fig. 2. Each bended-ellipse feature is represented by a vector which includes its coordinates, direction, angle, and arm length size. To save space, we skip the length portion in this work. This kind feature representation is much stable than the rigid skeleton representation. Each template outline is used to normalize the unknown by properly shifting, rotating, and scaling the unknown. The outline can be accomplished by thickening the template strokes.

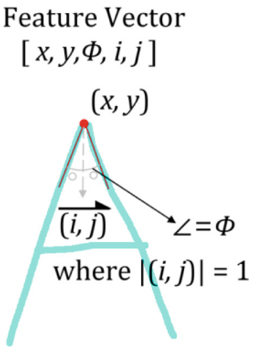

Fig. 2. The feature vector for the top portion of A.

Note the CNN [5] can extract local features in its front end layer as elements for constructions in its successive layers. High level correlations of such elements are picked in its deep layers. Recognition of unknown character can be accomplished from such correlations. Various distortions will increase its correlation complexity.

Representation by Bended-Ellipse Features

Suppose there are J templates $\{\mathbf{T}^1, \mathbf{T}^2, \ldots, \mathbf{T}^J\}$. Each template is represented by a collection of its all features. For each template $\mathbf{T}^j$ we will generate L^j bended-ellipse features. Write $\mathbf{T}^j = \{\mathbf{t}_l^j | 1 \le l \le L^j\}$ where $\mathbf{t}_l^j$ is the l^{th} feature vector of the j^{th} template. Let $\mathbf{t}_l^j = [x_l^j, y_l^j, \phi_l^j, i_l^j, j_l^j]$ be a five-dimensional row vector. The elements x_l^j and y_l^j in $\mathbf{t}_l^j$ are the x-y coordinates of the seed (corner) of the angle. Let (i_l^j, j_l^j) be the unit vector that extends from the seed and equally divides the angle ϕ_l^j, Fig. 1.

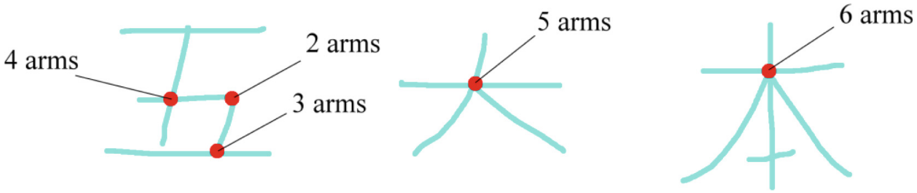

Fig. 3. Branch and cross.

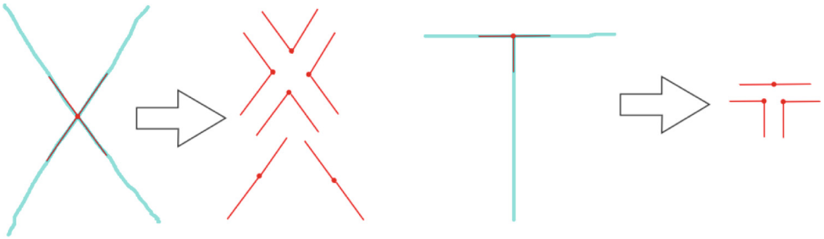

Fig. 4. Cross and branch points are represented by multiple features with a shared seed position.

The branch point and cross point will have three or more arms, Fig. 3. They will be represented by collections of bended-ellipse features with shared seed positions, Fig. 4. For example, a branch with 3 arms will be represented by 3 bended-ellipses and a cross with 4 arms will be represented by 6 bended-ellipses. A fork with more branches is important in recognition.

The feature collection that represents the target character, unknown handwritten character, is $\mathbf{U} = \{\mathbf{u}_m | 1 \le m \le M\}$, where $\mathbf{u}_m = [x_m, y_m, \phi_m, i_m, j_m]$ is the m^{th} feature vector of the unknown character.

2 Locking and Deforming Process

All templates and their various histograms must be well prepared in advance for candidate template screening. The unknown character must have more number of features, two to three times, than that of the template. To save computation, each template is carefully selected from the character corpus by the screening procedure.

It is expected that each selected template is additive to the recognition rate [10]. The template outline is used in the screening. The distribution of the template features in the i-j-ϕ diagram, Fig. 5, is also included as one related histogram.

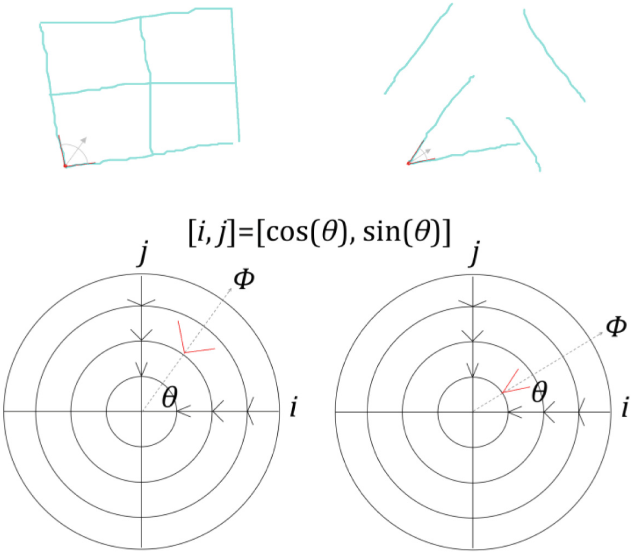

Fig. 5. Angle and direction histogram, i-j-ϕ diagram.

The similarity between two features, $\mathrm{s}\left(\mathbf{t}_l^j, \mathbf{u}_m\right)$, is defined as the inner product between the two features, $\mathbf{t}_l^j$ and $\mathbf{u}_m$. Set parameters that will be used in the following context, $\mathbf{t}_l^j = [x^t, y^t, \phi^t, i^t, j^t]$; $\mathbf{u}_m = [x^u, y^u, \phi^u, i^u, j^u]$;

$$\begin{aligned}\theta &= \cos^{-1}\left(\frac{\|(i^u,j^u)\|^2 + \|(i^t,j^t)\|^2 - \|(i^t,j^t) - (i^u,j^u)\|^2}{2\|(i^t,j^t)\|\|(i^u,j^u)\|}\right)\\ &= \cos^{-1}\left(\frac{2 - \|(i^t,j^t) - (i^u,j^u)\|^2}{2}\right);\end{aligned} \tag{1}$$

$$s_1 = \max\left\{0, \left(1 - \frac{\sqrt{(x^t - x^u)^2 + (y^t - y^u)^2}}{d}\right)\right\}; \tag{2}$$

$$s_2 = \max\left\{0, \left(1 - \frac{|\phi^t - \phi^u|}{\frac{\pi}{2}}\right)\right\};\ s_3 = \max\left\{0, \left(1 - \frac{|\theta|}{\frac{\pi}{2}}\right)\right\};\quad \text{and} \tag{3}$$

$$s_4 = \max\{0, (1 - |i^t i^u + j^t j^u|)\}. \tag{4}$$

$$\text{Then write } \mathrm{s}\left(\mathbf{t}_l^j, \mathbf{u}_m\right) = \begin{cases} \frac{1}{2}(s_1 + s_4), & \phi^t \geq \phi_{\text{line}} \\ \frac{1}{3}(s_1 + s_2 + s_3), & \textit{otherwise} \end{cases}.$$

$$\text{Note that } 0 \leq \mathrm{s}\left(\mathbf{t}_l^j, \mathbf{u}_m\right) \leq 1. \tag{5}$$

ϕ_{line} is a threshold to discriminate a straight line feature from a bended feature. We also use an alternate function in this work,

$$\mathrm{s}\left(\mathbf{t}_l^j, \mathbf{u}_m\right) = \min\left\{\mathrm{s}_\mathrm{p}\left(\mathbf{t}_l^j, \mathbf{u}_m\right), \mathrm{s}_\mathrm{d}\left(\mathbf{t}_l^j, \mathbf{u}_m\right)\right\}, \tag{6}$$

where s_p is the similarity between positions, s_d is the similarity between both angles and directions. Let

$$\mathrm{p}(\mathbf{v}) = \mathrm{p}([x, y, \phi, i, j]) = [x, y]. \tag{7}$$

Then the similarity of positions between two feature vectors is

$$\mathrm{s}_\mathrm{p}\left(\mathbf{t}_l^j, \mathbf{u}_m\right) = \mathrm{f}_\mathrm{p}\left(\frac{1}{w}\left\|\mathrm{p}\left(\mathbf{t}_l^j\right) - \mathrm{p}(\mathbf{u}_m)\right\|\right),$$

where

$$\begin{aligned} &\mathrm{f}_\mathrm{p}(x) = 2^{-8x^3} \text{ and} \\ &w = d_1 \, or \, d_2. \end{aligned} \tag{8}$$

The similarity between both angles and directions is

$$\mathrm{s}_\mathrm{d}(\mathbf{t}, \mathbf{u}) = \frac{1}{2} + \frac{1}{4}\max\{a_1(\mathbf{t}) \cdot a_1(\mathbf{u}) + a_2(\mathbf{t}) \cdot a_2(\mathbf{u}), a_1(\mathbf{t}) \cdot a_2(\mathbf{u}) + a_2(\mathbf{t}) \cdot a_1(\mathbf{u})\},$$

where

$$\begin{aligned} a_1(\mathbf{v}) = a_1([x, y, \phi, i, j]) = [i, j]\begin{bmatrix} \cos\frac{\phi}{2} & \sin\frac{\phi}{2} \\ -\sin\frac{\phi}{2} & \cos\frac{\phi}{2} \end{bmatrix} \text{ and} \\ a_2(\mathbf{v}) = a_2([x, y, \phi, i, j]) = [i, j]\begin{bmatrix} \cos\frac{-\phi}{2} & \sin\frac{-\phi}{2} \\ -\sin\frac{-\phi}{2} & \cos\frac{-\phi}{2} \end{bmatrix}. \end{aligned} \tag{9}$$

a_1 and a_2 are functions to reconstruct the direction vector of one arm of the feature. The whole similarity score between a template and the unknown is the sum of all similarities between every corresponding pairs, $\mathrm{S}\left(\mathbf{T}^j, \mathbf{U}\right) = \sum_{l=1}^{L^j} \mathrm{s}\left(\mathbf{t}_l^j, \mathbf{u}_l^j\right)$. Note that we set

$$\mathrm{s}\left(\mathbf{t}_l^j, \mathbf{u}_l^j\right) = \begin{cases} s\left(\mathbf{t}_l^j, \mathbf{u}_l^j\right) & \text{when there exists a matched } \mathbf{u}_l^j; \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

to include the case that $\mathbf{t}_l^j$ does not have a matched feature.

The process between the j^{th} template, and the unknown character, $\mathbf{U}$, is divided into three phases. In the first phase, each template feature, $\mathbf{t}_l^j \in \mathbf{T}^j$, finds its corresponding similar feature $\mathbf{u}_l^j$ in $\mathbf{U}$ within a small radius, d_1, where d_1 is a designed parameter, Fig. 6. In other words,

$$\mathbf{u}_l^j = \arg\max_{\mathbf{u}_m}\left\{\mathrm{s}\left(\mathbf{t}_l^j, \mathbf{u}_m\right)\right\} \text{ where } \left\|\mathrm{p}\left(\mathbf{t}_l^j\right) - \mathrm{p}(\mathbf{u}_m)\right\| < d_1. \tag{11}$$

In the second phases, pick the pair of the locked features, $\left(\mathbf{t}_{l_a}^j, \mathbf{u}_{l_a}^j\right)$, which is defined as

$$l_a = \arg\max_{l=1,..,L^j}\left\{\mathrm{s}\left(\mathbf{t}_l^j, \mathbf{u}_l^j\right)\right\}. \tag{12}$$

This locked pair receives the highest similarity value, $\mathrm{s}\left(\mathbf{t}_{l_a}^j, \mathbf{u}_{l_a}^j\right)$, among all pairs $\left\{\left(\mathbf{t}_l^j, \mathbf{u}_l^j\right),\ l = 1,..,L^j\right\}$. Then shift the whole template to a new position and overlap the seeds of these two features, $\mathbf{t}_{l_a}^j$ and $\mathbf{u}_{l_a}^j$, Fig. 7. Let $[x_a, y_a] = \mathrm{p}\left(\mathbf{u}_{l_a}^j\right)$, which is the seed position of the locked feature, $\mathbf{u}_{l_a}^j$. The shifted template, $\left(\mathbf{T}^j\right)^{\mathrm{s}}$, has new features $\left\{\left(\mathbf{t}_l^j\right)^{\mathrm{s}} = \left[\left(x_l^j\right)^{\mathrm{s}}, \left(y_l^j\right)^{\mathrm{s}}, \phi_l^j, i_l^j, j_l^j\right],\ l = 1, \ldots, L^j\right\}$, where $\left(x_l^j\right)^{\mathrm{s}} = x_l^j + \left(x_a - x_{l_a}^j\right)$ and $\left(y_l^j\right)^{\mathrm{s}} = y_l^j + \left(y_a - y_{l_a}^j\right)$. Then we repeat the first phase and find the corresponding features $\left(\mathbf{u}_l^j\right)^{\mathrm{s}}$ in $\mathbf{U}$ for every feature in the shifted template, $\left(\mathbf{t}_l^j\right)^{\mathrm{s}} \in \left(\mathbf{T}^j\right)^{\mathrm{s}}$. That is

$$\left(\mathbf{u}_l^j\right)^{\mathrm{s}} = \arg\max_{\mathbf{u}_m \in \mathbf{U}} \mathrm{s}\left(\left(\mathbf{t}_l^j\right)^{\mathrm{s}}, \mathbf{u}_m\right) \text{ where } \left\|\mathrm{p}\left(\left(\mathbf{t}_l^j\right)^{\mathrm{s}}\right) - \mathrm{p}(\mathbf{u}_m)\right\| < d_2. \tag{13}$$

Let $\left\{\left(\left(\mathbf{t}_l^j\right)^{\mathrm{s}}, \left(\mathbf{u}_l^j\right)^{\mathrm{s}}\right),\ l = 1,..,L^j\right\}$ be the corresponding pairs between the shifted template $\left(\mathbf{T}^j\right)^{\mathrm{s}}$ and unknown $\mathbf{U}$. The new whole similarity score is $\mathrm{S}\left(\left(\mathbf{T}^j\right)^{\mathrm{s}}, \mathbf{U}\right) = \sum_{l=1}^{L^j} \mathrm{s}\left(\left(\mathbf{t}_l^j\right)^{\mathrm{s}}, \left(\mathbf{u}_l^j\right)^{\mathrm{s}}\right)$. Note that the locked pairs, $\left(\mathbf{t}_{l_a}^j, \mathbf{u}_{l_a}^j\right)$, is in some sense similar to the resonance portion in ART [12].

In the third phase, pick the second locked feature pair that received the highest similarity value, $\left(\left(\mathbf{t}_{l_b}^j\right)^{\mathrm{s}}, \left(\mathbf{u}_{l_b}^j\right)^{\mathrm{s}}\right)$, where l_b is not equal to l_a,

$$l_b = \arg\max_{l=1,..,L^j}\left\{\mathrm{s}\left(\left(\mathbf{t}_l^j\right)^{\mathrm{s}}, \left(\mathbf{u}_l^j\right)^{\mathrm{s}}\right)\right\}, \text{ and } \left(\mathbf{u}_{l_b}^j\right)^{\mathrm{s}} \neq \mathbf{u}_{l_a}^j \tag{14}$$

Let $[x_b, y_b] = \mathrm{p}\left(\left(\mathbf{u}_{l_b}^j\right)^{\mathrm{s}}\right)$ be the seed position of the feature $\left(\mathbf{u}_{l_b}^j\right)^{\mathrm{s}}$. Using the position $[x_a, y_a]$ as the rotation center and the scaling center, rotate and scale the whole template $\left(\mathbf{T}^j\right)^{\mathrm{s}}$ so that the two seeds $\mathrm{p}\left(\left(\mathbf{t}_{l_a}^j\right)^{\mathrm{s}}\right) = [x_a, y_a]$ and $\mathrm{p}\left(\left(\mathbf{t}_{l_b}^j\right)^{\mathrm{s}}\right) = [x_b, y_b]$ will

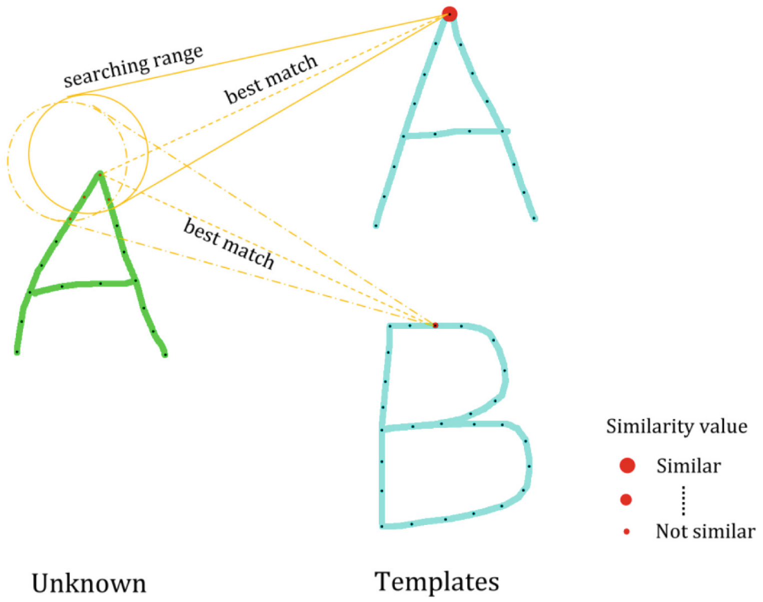

Fig. 6. Searching corresponding features in unknown character.

coincide with their corresponding seeds in the unknown character. After rotation and scaling, we obtain a refined template collection, $\left(\mathbf{T}^j\right)^{\mathrm{r}}$, where

$$\left(\mathbf{T}^j\right)^{\mathrm{r}} = \left\{\left(\mathbf{t}_l^j\right)^{\mathrm{r}} = \left[\left(x_l^j\right)^{\mathrm{r}}, \left(y_l^j\right)^{\mathrm{r}}, \phi_l^j, \left(i_l^j\right)^{\mathrm{r}}, \left(j_l^j\right)^{\mathrm{r}}\right], l = 1, \ldots, L^j\right\} \tag{15}$$

In order to facilitate the operation of the rotation and scaling procedure for the whole template, set complex numbers,

$$\begin{gathered} \operatorname{cpx}(\mathbf{v}) = \operatorname{cpx}([x, y]) = [x, y] \cdot [1, i] = x + yi, \\ \operatorname{vec}(x + yi) = [x, y] = \mathbf{v}. \end{gathered} \tag{16}$$

Then, operate the rotation and scaling according to the seed positions of the two locked feature pairs. Set

$$\mathbf{p}_{\mathrm{t}} = \mathrm{p}\left(\left(\mathbf{t}_{l_b}^j\right)^{\mathrm{s}}\right) - \mathrm{p}\left(\mathbf{u}_{l_a}^j\right) \text{ and } \mathbf{p}_{\mathrm{u}} = \mathrm{p}\left(\left(\mathbf{u}_{l_b}^j\right)^{\mathrm{s}}\right) - \mathrm{p}\left(\mathbf{u}_{l_a}^j\right).$$

Also set

$$c_{\mathrm{r,s}} = \frac{\operatorname{cpx}(\mathbf{p}_{\mathrm{u}})}{\operatorname{cpx}(\mathbf{p}_{\mathrm{t}})} \quad \text{and} \quad c_{\mathrm{r}} = \frac{\operatorname{cpx}\left(\frac{\mathbf{p}_{\mathrm{u}}}{|\mathbf{p}_{\mathrm{u}}|}\right)}{\operatorname{cpx}\left(\frac{\mathbf{p}_{\mathrm{t}}}{|\mathbf{p}_{\mathrm{t}}|}\right)}.$$

After we get these complex numbers, apply them to the position vector and the direction vector of every template features. Then, obtain a rotated and scaled template,

$$\begin{aligned}&\left[\left(x_l^j\right)^{\mathrm{r}},\left(y_l^j\right)^{\mathrm{r}}\right]=\operatorname{vec}\left(\operatorname{cpx}\left(\mathrm{p}\left(\left(\mathbf{t}_l^j\right)^{\mathrm{s}}\right)-\mathrm{p}\left(\mathbf{u}_{l_a}^j\right)\right)\times c_{\mathrm{r,s}}\right)\text{ and}\\&\left[\left(i_l^j\right)^{\mathrm{r}},\left(j_l^j\right)^{\mathrm{r}}\right]=\operatorname{vec}\left(\operatorname{cpx}\left(\left[\left(i_l^j\right)^{\mathrm{s}},\left(j_l^j\right)^{\mathrm{s}}\right]\right)\times c_{\mathrm{r}}\right),\quad\text{for }\forall\left(\mathbf{t}_l^j\right)^{\mathrm{s}}\in\left(\mathbf{T}^j\right)^{\mathrm{s}},\end{aligned}\tag{17}$$

where $\mathrm{p}\left(\left(\mathbf{t}_{l_a}^j\right)^{\mathrm{r}}\right)=\mathrm{p}\left(\mathbf{u}_{l_a}^j\right)$ and $\mathrm{p}\left(\left(\mathbf{t}_{l_b}^j\right)^{\mathrm{r}}\right)=\mathrm{p}\left(\left(\mathbf{u}_{l_b}^j\right)^{\mathrm{s}}\right)$.

Repeating the procedure in the first phase for all features, $\forall\left(\mathbf{t}_l^j\right)^{\mathrm{r}}\in\left(\mathbf{T}^j\right)^{\mathrm{r}}$, get

$$\left(\mathbf{u}_l^j\right)^{\mathrm{r}}=\arg\max_{\mathbf{u}_m\in\mathbf{U}}\mathrm{s}\left(\left(\mathbf{t}_l^j\right)^{\mathrm{r}},\mathbf{u}_m\right),\text{ where }\left\|\mathrm{p}\left(\left(\mathbf{t}_l^j\right)^{\mathrm{r}}\right)-\mathrm{p}(\mathbf{u}_m)\right\|<d_3,\tag{18}$$

The refined whole score for the refined template is $\mathrm{S}\left(\left(\mathbf{T}^j\right)^{\mathrm{r}},\mathbf{U}\right)=\sum_{l=1}^{L^j}\mathrm{s}\left(\left(\mathbf{t}_l^j\right)^{\mathrm{r}},\left(\mathbf{u}_l^j\right)^{\mathrm{r}}\right)$.

The three scores, $\mathrm{S}\left(\mathbf{T}^j,\mathbf{U}\right)$, $\mathrm{S}\left(\left(\mathbf{T}^j\right)^{\mathrm{s}},\mathbf{U}\right)$, and $\mathrm{S}\left((\mathbf{T})^{\mathrm{r}},\mathbf{U}\right)$, will be used as the compatibility scores for the j^{th} template. Pick the largest value of the three as the final compatibility for the template $\mathbf{T}^j$,

$$\mathrm{S}^*\left(\mathbf{T}^j,\mathbf{U}\right)=\max\left\{\mathrm{S}\left(\mathbf{T}^j,\mathbf{U}\right),\mathrm{S}\left(\left(\mathbf{T}^j\right)^{\mathrm{s}},\mathbf{U}\right),\mathrm{S}\left(\left(\mathbf{T}^j\right)^{\mathrm{r}},\mathbf{U}\right)\right\}.\tag{19}$$

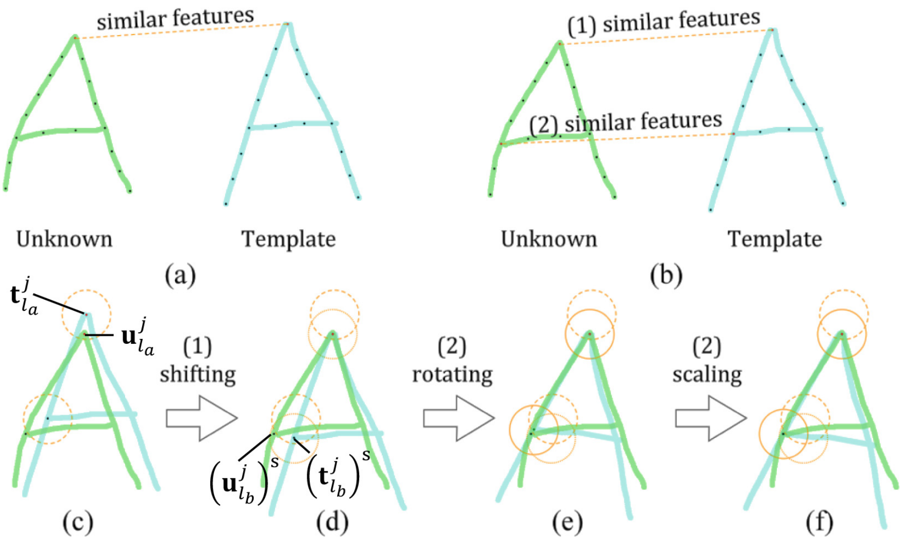

Fig. 7. The two seeds of the two locked features will be aligned perfectly after shifting, rotation, and scaling procedures.

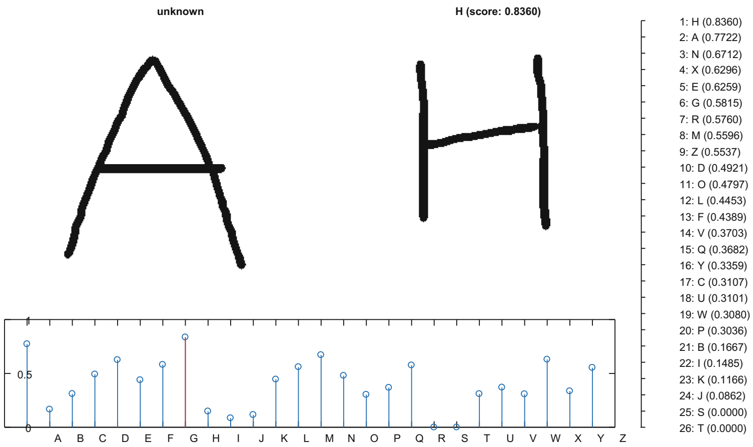

Fig. 8. The unknown character ‘A’, 256 by 256 pixels.

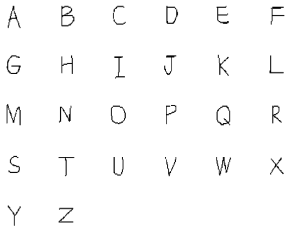

Fig. 9. Images of the 26 templates.

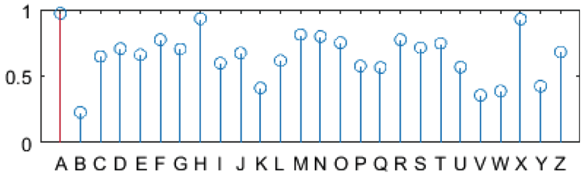

Fig. 10. Similarity scores (Eq. 7).

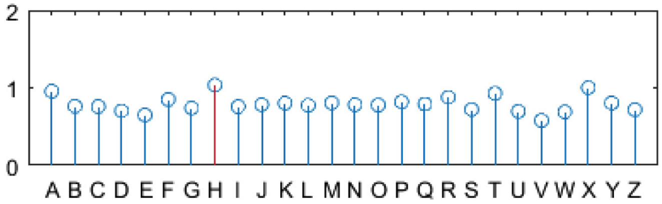

Fig. 11. Similarity scores (Eq. 6).

3 Simulation and Discussion

We tested the process to examine its capability. In our simulation, the size of the image of each character is 256-by-256 pixels for both templates and the unknown character, Fig. 8. The whole scores are also listed in the figure. All 26 templates are the 26 images of handwritten English uppercase alphabets, Fig. 9. The width of each stroke is roughly 5 to 15 pixels. Each image is properly normalized. The similarity scores, Eq. 7, of all 26 templates are listed in Fig. 10. The similarity scores, Eq. 6, are listed in Fig. 11.

The recognition rate of the proposed process is comparable to that of STD. The computation cost is much less than that of STD. Since each template is designed manually, several fixed key features can be specified manually in advance. These keys have priority in the process. The highest locked pair $\left(\mathbf{t}^j_{l_a}, \mathbf{u}^j_{l_a}\right)$ may be picked only from those keys. The second highest pair l_b, $\left(\left(\mathbf{t}^j_{l_b}\right)^{\mathrm{S}}, \left(\mathbf{u}^j_{l_b}\right)^{\mathrm{S}}\right)$, may also be picked from them. The whole scores S may be accumulated only for those key features.

The two locked pairs can be picked simultaneously in the first phase. When one picked three different locked pairs, one can apply the conformal mapping technique [8] to deform the whole template and accomplish the score.

References

1. Blasdel, G.G., Salama, G.: Voltage-sensitive dyes reveal a modular organization in monkey striate cortex. Nature **321**, 579–585 (1986)
2. Carpenter, G.A., Grossberg, S.: ART 2: self-organization of stable category recognition codes for analog input patterns. Appl. Opt. **26**(23), 4919–4930 (1987)
3. Hubel, D.H., Wiesel, T.N.: Brain mechanisms of vision. In: Rock, I. (ed) The Perceptual World: Readings from Scientific American Magazine, pp. 3–24 (1990)
4. Kohonen, T.: Self-Organization and Associative Memory, 3rd edn. Springer, Berlin (1989). https://doi.org/10.1007/978-3-642-88163-3
5. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
6. Liou, D.-R., Lin, C.-C., Liou, C.-Y.: Setting shape rules for handprinted character recognition. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ACIIDS 2012. LNCS (LNAI), vol. 7197, pp. 245–252. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28490-8_26
7. Liou, C.-Y., Shih, H.-Y., Liou, D.-R.: Finite geometrical relations loading in Hopfield model. J. Theor. Appl. Comput. Sci. **6**(4), 59–76 (2012)
8. Liou, C.-Y., Tai, W.-P.: Conformal self-organization for continuity on a feature map. Neural Netw. **12**, 893–905 (1999)
9. Liou, C.-Y., Yang, H.-C.: Spatial topology distance for handprinted character recognition. In: Gielen, S., Kappen, B. (eds.) ICANN 1993, pp. 918–921. Springer, London (1993). https://doi.org/10.1007/978-1-4471-2063-6_266
10. Liou, C.-Y., Yang, H.C.: Handprinted character recognition based on spatial topology distance measurement. IEEE Trans. Pattern Anal. Mach. Intell. **18**(9), 941–945 (1996)

11. Liou, C.Y., Yang, H.C.: Selective feature-to-feature adhesion for recognition of cursive handprinted characters. IEEE Trans. Pattern Anal. Mach. Intell. **21**(2), 184–191 (1999)
12. Nasrabadi, N.M., Li, W., Choo, C.Y.: Object recognition by a Hopfield neural network. In: ICCV, pp. 325–328 (1990)
13. Suganthan, P.N., Teoh, E.K., Mital, D.P.: Pattern recognition by homomorphic graph matching using Hopfield neural networks. Image Vis. Comput. **13**(1), 45–60 (1995)