Feature Desynchronization in Online Character Recognition

Seiichi Uchida *Kazuya Niyagawa* *Hiroaki Sakoe*
Kyushu University
744 Motooka, Nishi-ku, Fukuoka-shi, 819-0395, Japan
uchida@is.kyushu-u.ac.jp

Abstract

This paper discusses desynchronization of feature elements, such as X-coordinate and Y-coordinate, in online character recognition. The main results were twofold: First, the feature desynchronization has ability to compensate the shape difference between character patterns. Second, the feature desynchronization is useful to improve recognition accuracy if its range is regulated according to actual deformation characteristics of characters.

Keywords: desynchronization, feature vector, online character recognition, elastic matching

1. Introduction

In online character recognition, each character pattern is usually represented as a temporal sequence of feature vectors. As the elements of the feature vector, X- and Y-coordinates, local direction, pen pressure, and pen tilt have been utilized. Those elements are generally acquired at the same timing and therefore used synchronously. For example, if a character pattern is represented as a sequence of two-dimensional coordinate feature vectors, i.e., $(X_1, Y_1), \ldots, (X_j, Y_j), \ldots, (X_J, Y_J)$, the elements X_j and X_j are always coupled tightly and not treated in any independent manner. Note that this synchronous usage of the feature elements is also common in other temporal pattern recognition tasks, such as speech recognition and gesture recognition.

The main purpose of this paper is to see the effect of *desynchronization* of feature elements in online character recognition. Under the desynchronization, two feature elements acquired at the different timings can be coupled as a virtual feature vector like (X_j, Y_k).

Note that the desynchronization discussed in this paper does not mean that feature elements are used in a *totally* independent manner. In fact, the range of the desynchronization should be limited because the totally independent desynchronization will cause unexpected results as experimentally shown in a later section.

As the first trial of desynchronization in online character recognition, we assume the above two-dimensional coordinate feature vector (X_j, Y_j). In other words, we assume that horizontal pen movement and vertical pen movement are partially asynchronous on writing character patterns. Although this assumption has not been fully supported by any theory at this moment, our trial is still meaningful because of the following points. First, the desynchronization of (X_j, Y_j) has the ability of synthesizing new patterns from a single pattern (Section 4) and this ability is useful to achieve better recognition performance (Section 5). Second, we can find some hint of the theoretical support in the past research on writing kinematics (Section 6.2). Third, it is possible to extend this trial to a more solid trial by using other feature elements (Section 6.1).

In this paper, the feature desynchronization is examined in a online character recognition technique based on elastic matching. Elastic matching provides an optimal point-to-point correspondence between two character patterns and therefore can not only compensate nonlinear temporal fluctuation but also adjust variations in pattern length. For the optimization, we will employ dynamic programming (DP), which has been widely used (e.g., [1, 2]).

2. Related work

There are few past attempts on feature desynchronization in sequential pattern recognition tasks [1] other than online character recognition. Those attempts have tried to couple two hidden Markov models (HMMs), which can be considered as a statistical extension of the DP-based elastic matching technique used in this paper. Brand et al. [3] has coupled two HMMs for gesture recognition. A similar idea can be found in Matsuda et al. [4], where coupled HMMs has been applied to speech recognition.

This paper is different from those past attempts at several points; for example, the difference in models (DP and HMM), the difference in applications, and the difference

[1] It is interesting to note that the desynchronized matching algorithm has a close relation to the algorithms in genome science as discussed in Section 6.3.

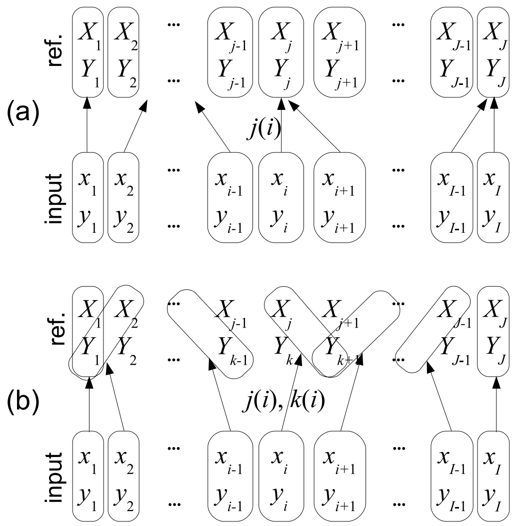

Figure 1. (a) Conventional elastic matching. (b) Desynchronized matching.

in features. In addition to those superficial differences, there is an essential difference that the purpose of this paper includes the observation on the effect of the desynchronization. In fact, the desynchronization has a powerful effect of synthesizing virtual patterns (Section 4.2), while the effect has not been pointed out in the past attempts.

Artières et al. [5] also have proposed several algorithms of coupled HMMs and applied them to online character recognition. They have coupled an HMM for online recognition with an HMM for offline recognition. From an algorithmic viewpoint, none of their algorithms is the same as the proposed one. It will be possible to apply the proposed algorithm to their task for more accurate evaluation with less complexity. In addition, there is the above essential difference with each other.

3. Conventional elastic matching technique

3.1. Problem formulation

Let $\boldsymbol{E} = \boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_i, \ldots, \boldsymbol{e}_I$ denote an input pattern and $\boldsymbol{R}_c = \boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_j, \ldots, \boldsymbol{r}_J$ denote the reference pattern of the category c. (Strictly speaking, $\boldsymbol{r}_j$ and J should be denoted as $\boldsymbol{r}_{c,j}$ and J_c.) Throughout this paper, we will assume the two-dimensional coordinate feature vector, i.e., $\boldsymbol{e}_i = (x_i, y_i)^T, \boldsymbol{r}_j = (X_j, Y_j)^T$.

The nature of elastic matching is to establish the optimal point-to-point correspondence between $\boldsymbol{E}$ and $\boldsymbol{R}_c$. The matching cost under the optimal correspondence is invariant to temporal fluctuation and therefore useful to realize practical recognition systems.

The conventional DP-based elastic matching technique is formulated as the following optimal correspondence problem between $\boldsymbol{E}$ and $\boldsymbol{R}_c$:

$$\begin{aligned} \mathcal{J}_{\rm sync} &= \sum_{i=1}^{I} \|\boldsymbol{e}_i - \boldsymbol{r}_{j(i)}\| \\ &= \frac{1}{I}\sum_{i=1}^{I}\sqrt{(x_i - X_{j(i)})^2 + (y_i - Y_{j(i)})^2}, \quad (1) \end{aligned}$$

where the sequence $j(1), \ldots, j(i), \ldots, j(I)$ represents the point-to-point correspondence to be optimized. At the optimization, we assume the boundary constraints, $j(1) = 1$ and $j(I) = J$, and the so-called monotonicity and continuity constraint,

$$j(i) - j(i-1) \in \{0, 1, 2\}. \quad (2)$$

It is important to note that two feature elements are used synchronously in (1), that is, X and Y are equally indexed by $j(i)$. Accordingly, as shown in **Fig. 1** (a), the two elements x_i and y_i at the same ith point on $\boldsymbol{E}$ correspond to the elements at the same $j(i)$th point on $\boldsymbol{R}_c$.

3.2. Algorithm of conventional elastic matching

The optimal correspondence problem of (1) can be considered as an optimal path problem on the $i - j$ plane. **Fig. 2** illustrates the optimal path representing $j(1), \ldots, j(i), \ldots, j(I)$. The constraint (2) assures a smoothness and the monotonicity of the path.

This optimal path problem can be solved efficiently by a DP-based algorithm. Letting $d(i,j) = \sqrt{(x_i - X_j)^2 + (y_i - Y_j)^2}$, the DP algorithm is organized as calculation of the following recursive equation at all j from $i = 1$ to I:

$$g(i,j) = d(i,j) + \min_{j' \in \{j-2, j-1, j\}} g(i-1, j'), \quad (3)$$

where $g(i,j)$ represents the minimum matching cost between $\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_i$ and $\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_j$. Consequently, $g(I,J)$ is the minimum matching cost between $\boldsymbol{E}$ and $\boldsymbol{R}_c$, i.e.,

$$\min \mathcal{J}_{\rm sync} = g(I,J), \quad (4)$$

and invariant to temporal fluctuation of $\boldsymbol{E}$. The minimum matching cost $g(I,J)$ is used as a discrimination function; the category c giving the smallest $g(I,J)$ is selected as the recognition result of $\boldsymbol{E}$.

4. Desynchronized matching

4.1. Problem formulation

The elastic matching problem with the desynchronization of X–Y feature elements is formulated by slightly modifying the conventional elastic matching problem as follows:

$$\mathcal{J}_{\rm desync} = \sum_{i=1}^{I}\sqrt{(x_i - X_{j(i)})^2 + (y_i - Y_{k(i)})^2}, \quad (5)$$

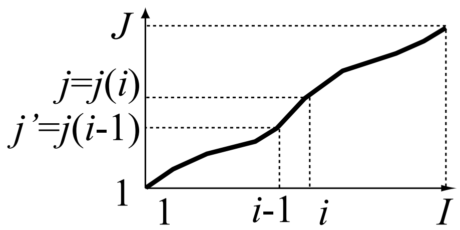

Figure 2. Conventional elastic matching as an optimal path problem.

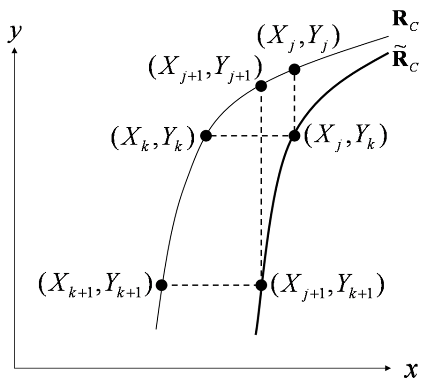

Figure 3. Pattern synthesis by the desynchronization of X–Y feature elements.

where $j(i)$ and $k(i)$ represent the point-to-point correspondences on x and y coordinate feature elements, respectively. The objective function $\mathcal{J}_{\rm desync}$ is to be minimized with respect to $j(1),\ldots,j(i),\ldots,j(I)$ and $k(1),\ldots,k(i),\ldots,k(I)$ under the boundary conditions $j(1)=k(1)=1$ and $j(I)=k(I)=J$ and the monotonicity and continuity constraints,

$$\left\{\begin{array}{l} j(i)-j(i-1)\in\{0,1,2\},\\ k(i)-k(i-1)\in\{0,1,2\}.\end{array}\right. \qquad (6)$$

Fig. 1 (b) depicts the idea of the desynchronization of x and y-coordinate feature elements. Under the desynchronization, those feature elements of the ith point on $\boldsymbol{E}$ correspond to different points, $j(i)$ and $k(i)$, on $\boldsymbol{R}$ respectively. In other words, $\boldsymbol{e}_i=(x_i,y_i)^T$ corresponds to $\tilde{\boldsymbol{r}}_i=(X_{j(i)},Y_{k(i)})^T$.

4.2. Effect of desynchronization

It is an important fact that the desynchronization of X–Y feature elements is a pattern synthesis process. As shown in **Fig. 3**, $\tilde{\boldsymbol{R}}_c=\tilde{\boldsymbol{r}}_1,\ldots,\tilde{\boldsymbol{r}}_i,\ldots,\tilde{\boldsymbol{r}}_I$ is a synthetic pattern, since each point $\tilde{\boldsymbol{r}}_i=(X_{j(i)},Y_{k(i)})^T$ is not an actual point on $\boldsymbol{R}$. Accordingly, if we control the point correspondence $(j(1),k(1)),\ldots,(j(i),k(i)),\ldots,(j(I),k(I))$, we can produce various synthetic patterns. The ability of synthesizing patterns is very impressive; from a diagonal line segment, we can synthesize any monotonic and continuous curve connecting the two ends of the original line segment, if we allow free desynchronization.

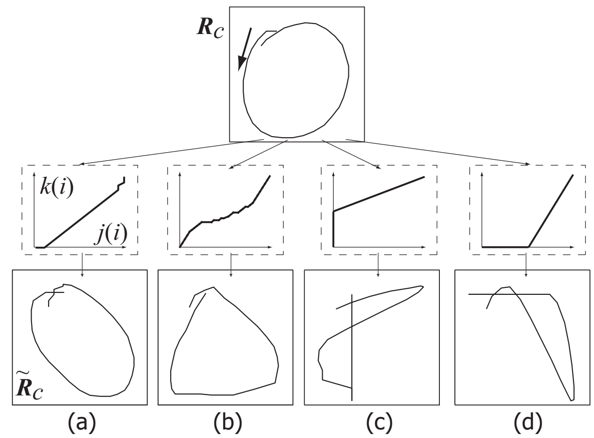

Figure 4. Synthetic patterns $\tilde{\boldsymbol{R}}_c$ by desynchronization.

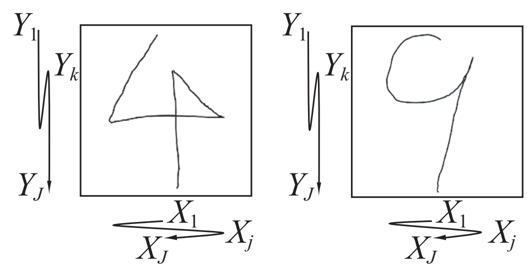

Figure 5. A pair of character patterns with similar horizontal and vertical pen movements.

Although the desynchronization is a "tricky" way to synthesize deformed patterns from a single reference pattern, it has a high potential to mimic actual patterns. In fact, **Figs. 4** (a) and (b) are synthetic patterns from the same pattern "0" by different desynchronized correspondences. Those synthetic patterns seem natural (actually, we found a sample like (b) in the database used) and thus the desynchronized matching technique is more tolerant to the deformations than the conventional elastic matching technique which compensates temporal difference only.

On the other hand, it is possible to provide unexpected synthetic patterns as shown in **Figs. 4** (c) and (d). Such unexpected patterns will degrade recognition accuracy because they may result in the misrecognition due to *overfitting*, which is the phenomenon that the reference pattern of a wrong category is closely fitted to the input pattern.

In fact, a character pair of "4" and "9" of **Fig. 5**

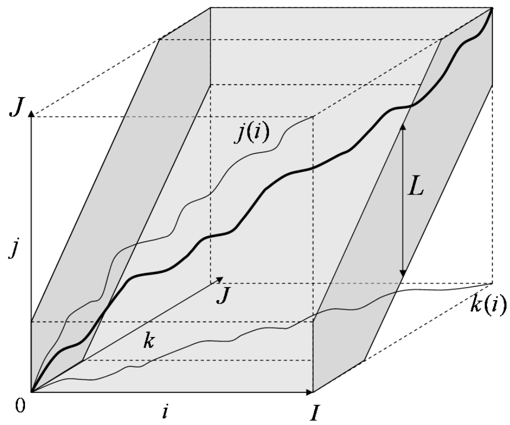

Figure 6. Desynchronized matching (with constant lag limitation) as an optimal path problem.

are easily misrecognized by free desynchronization. The horizontal movement of "4" is almost the same as that of "9" and therefore $x_i \sim X_{j(i)}$ under the optimized correspondence $j(1),\ldots,j(i),\ldots,j(I)$. Similarly, $y_i \sim Y_{k(i)}$ under the optimized correspondence $k(1),\ldots,k(i),\ldots,k(I)$. Consequently, the matching cost $\min \mathcal{J}_{\mathrm{desync}}$ becomes very small between "4" and "9" by an unexpected (i.e., over-fitted) synthetic pattern.

4.3. Limiting desynchronization

It is possible to avoid unexpected synthetic patterns by limiting the range of desynchronization. Two limitation schemes, constant lag limitation and adaptive lag limitation, are introduced in the following.

4.3.1. Constant lag limitation

One of the simplest ways to limit the desynchronization is to introduce the following constraint on the correspondence:

$$\|k(i) - j(i)\| \leq L/2, \tag{7}$$

where L is a positive constant to limit the maximum *lag* between $j(i)$ and $k(i)$. Its effect is shown in **Fig. 4** where (a) and (b) are synthetic patterns with the limitation at $L = 10$ whereas (c) and (d) are those without limitation. Note that if $L = 0$, the desynchronized matching is reduced to the conventional elastic matching technique of Section 3.

4.3.2. Adaptive lag limitation

The constant lag limitation (7) assumes that the maximum lag L is independent of j and k values. In other words, the deformation range of the character $\boldsymbol{R}_c$ is assumed not to change with the positions on the character stroke. This assumption, however, is somewhat rough. This is because every character may have its local deformation characteristics; heavy deformations appear around some position and not around another position. For example, the deformations at the beginning part of "1" are heavier than those at its horizontal ending part.

The following adaptive lag limitation is a simple and reasonable extension of the constant lag limitation to deal with the local deformation characteristics:

$$\|k(i) - j(i)\| \leq L_{j(i)}/2, \tag{8}$$

where the positive parameter L_j changes with j. The parameter L_j must be optimized for each category c in some way. (Thus, strictly speaking, L_j must be denoted as $L_{c,j}$.) An example of the optimization will be given in Section 5.3.

The optimized parameter L_j will reflect local deformation characteristics of the category c. A larger (smaller) L_j reflects larger (smaller) deformations around j. Thus, L_j around the beginning part of "1" may become a large value.

4.4. Algorithm of desynchronized matching

The optimal correspondence problem of (5) can be considered as the optimal path problem on the three-dimensional i-j-k space and still can be solved by a DP-based algorithm. **Fig. 6** illustrates the DP algorithm which searches for the optimal path representing $(j(1),k(1)),\ldots,(j(i),k(i)),\ldots,(j(I),k(I))$ under the lag limitation, (7) or (8).

Letting $d(i,j,k) = \sqrt{(x_i - X_j)^2 + (y_i - Y_k)^2}$, the DP algorithm for the desynchronized matching is organized as the calculation of the following recursive equation at all pairs of (j,k) from $i = 1$ to I:

$$g(i,j,k) = d(i,j,k) + \min_{\substack{j' \in \{j-2,j-1,j\} \\ k' \in \{k-2,k-1,k\}}} g(i-1,j',k'). \tag{9}$$

If the lag limitation is introduced, the equation (9) must not be calculated at any pair (j,k) violating the limitation. The matching cost $\min \mathcal{J}_{\mathrm{desync}}$ is provided as $g(I,J,J)$.

5. Experimental results

5.1. Character samples

The numeric character samples from the public database called Ethem Alpaydin Digit [6] were used in the experiment. The database is comprised of 7,494 training samples and 3,498 test samples. The training samples were written by 30 writers and the test samples were written by other 14 writers. The training samples were used only for training the parameters of the adaptive lag limitation in Section 5.3. Every sample was preprocessed to be a single-stroke character by simply connecting its component strokes and then linearly rescaled to 128×128

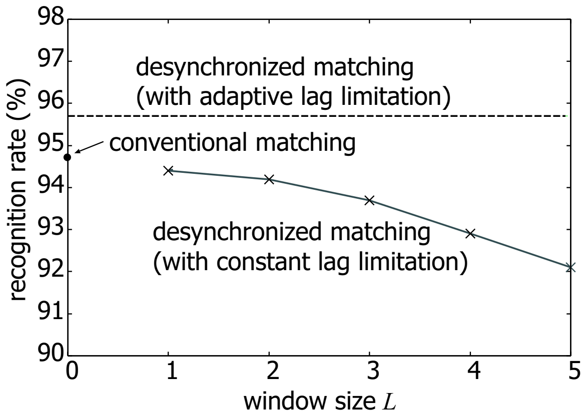

Figure 7. Recognition accuracy.

while keeping its original aspect ratio. Reference patterns $\{\boldsymbol{R}_c\}$ were created manually to cover typical writing orders. The number of reference patterns was 21; $1 \sim 4$ reference patterns were prepared for each category.

5.2. Recognition results with constant lag limitation

Fig. 7 plots the recognition accuracies by the conventional (synchronous) elastic matching technique and the proposed desynchronized elastic matching technique under the constant lag limitation (7) at $L = 1, \ldots, 5$. Note again that the proposed technique at $L = 0$ is reduced to the conventional technique. The conventional technique achieved the accuracy of 94.7%[2].

As shown in **Fig. 7**, the proposed technique with the constant lag limitation could not outperform the conventional technique at any L. Moreover, a larger L achieved a lower recognition accuracy. **Fig. 8** shows a typical misrecognition result. Under the constant lag limitation, the reference pattern $\boldsymbol{R}_c$ of the category "2" was over-fitted to the input pattern $\boldsymbol{E}$ of "1" as $\tilde{\boldsymbol{R}}_c$. In fact, curved parts of "2" became diagonal lines by the desynchronization.

5.3. Recognition results with adaptive lag limitation

Another experiment has been conducted to show that the recognition accuracy can be improved by the adaptive lag limitation. In this paper, the maximum lag L_j was parameterized by $L1$, $L2$, $L3$, $B1$, and $B2$, as shown in **Fig. 9**. Specifically,

$$L_j = \begin{cases} L1, & 1 \leq j \leq B1, \\ L2, & B1 \leq j \leq B2, \\ L3, & B2 \leq j \leq J. \end{cases} \tag{10}$$

[2]We can expect a higher recognition rate by another feature vector, such as (x_i, y_i, θ_i) where θ_i is local writing direction. Since the purpose of this paper is to observe the effect of the desynchronization, we adhere to the simpler feature vector (x_i, y_i).

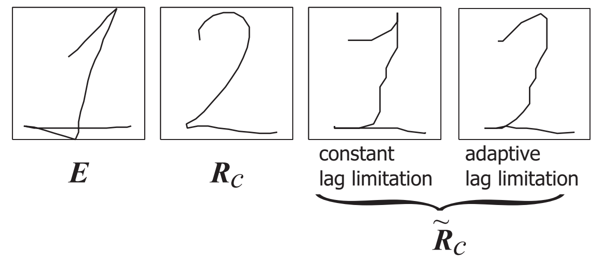

Figure 8. A pair of patterns $\boldsymbol{E}$ and $\boldsymbol{R}_c$ and their matching results $\tilde{\boldsymbol{R}}_c$.

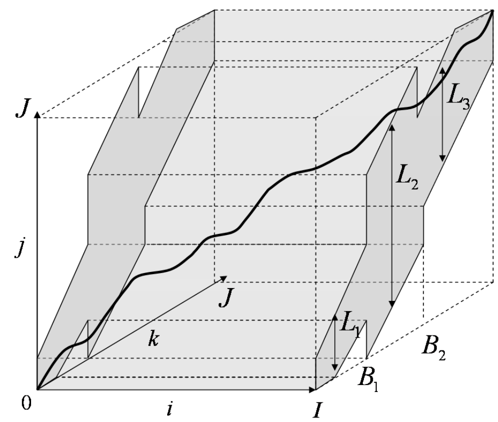

Figure 9. Desynchronized matching with adaptive lag limitation.

For each category c, those five parameters were optimized in an exhaustive manner. The parameter were fixed at the values which gave the highest recognition accuracy for the training patterns of the category.

Fig. 10 shows the optimized parameters of the adaptive lag limitation for the categories "0", "1", and "2." For the category "0", the optimal values of $L1$, $L2$, and $L3$ were found equally at 2. This fact indicates that the local deformation characteristics of "0" are uniform. The parameters for "1" coincide with the fact that its beginning part tends to be deformed more heavily than those at its ending part.

The recognition accuracy achieved with the adaptive lag limitation was 95.7% and therefore the proposed technique could outperform the conventional technique. As shown in **Fig. 8**, the over-fitting of "2" to "1" was suppressed successfully by the adaptive lag limitation.

6. Discussion

6.1. Desynchronization of other features

Desynchronization between the coordinate feature and a pen-up/down feature is another promising trial. Desyn-

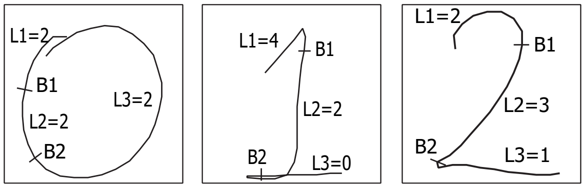

Figure 10. Adaptive lag limitation for the categories "0", "1", and "2."

chronization of those features will appear as a hook or a shortened stroke and therefore be useful to compensate those deformations. The result of this trial will be discussed elsewhere.

On the other hand, we must be careful in the desynchronization of feature elements with strong mutual dependence. For example, the desynchronization of the coordinate feature (X_j, Y_j) and the local pen direction feature θ_j is not a simple problem because θ_j is generally derived by (X_j, Y_j) and (X_{j-1}, Y_{j-1}) and thus thoughtless desynchronization will cause conflicting results. (Actually, we cannot have a consistent synthetic pattern by the desynchronization.)

6.2. Relation to the theory of writing kinematics

At this moment, there is no firm theoretical support of the deformation compensation by the desynchronization of X–Y feature elements, as noted in Section 1. We, however, can find some relation to the well-known writing model by Flash and Hogan [7]. In their writing model, two diagonally aligned points are connected by a minimum-jerk curve whereas two vertically (or horizontally) aligned points are connected by a straight line. This property resembles synthetic patterns by desynchronization and may support the validity of the desynchronization matching.

6.3. Relation to multiple alignment problem

From an algorithmic viewpoint, the DP-based algorithm for desynchronized matching in Section 4.4 are closely related to the so-called multiple alignment problem, which has been investigated actively in genome science[8]. If we want to desynchronize three or more feature elements, the computational complexity of the optimal path problem becomes intractable. Thus, we should resort to various approximation algorithms developed for the multiple alignment problems.

7. Conclusion

Desynchronization of feature elements was proposed and applied to online character recognition. The feature desynchronization had ability to compensate the shape difference between character patterns by combining feature elements at different timings. It was shown experimentally that the desynchronization can improve recognition accuracy (94.7%→ 95.7% for numerics) by incorporating actual deformation characteristics as a limitation of the desynchronization.

Future work will focus on the desynchronization of other feature elements as noted in Section 6.1. Investigating theoretical relation to the writing kinematics will be an interesting topic. We can apply the idea of the feature desynchronization to other sequential patterns, such as gesture patterns. When each gesture pattern is represented as a sequence of two-dimensional feature vectors representing the movements of the left and the right hands, we can expect that the virtual feature vector by the desynchronization can compensate the asynchronous temporal distortions of the hands. Patterns acquired from multimodal channels also can be promising application.

Acknowledgment: This work is partially supported by Microsoft Research Asia Mobile Computing in Education Theme Program.

References

[1] S. D. Connell and A. K. Jain, "Template-based online character recognition," Pattern Recognit., vol. 34, no. 1, pp. 1–14, 2001.

[2] C. Bahlmann and H. Burkhardt, "The writer independent online handwriting recognition system *frog on hand* and cluster generative statistical dynamic time warping," IEEE Trans. PAMI, vol. 26, no. 3, pp. 299-310, 2004.

[3] M. Brand, N. Oliver, and A. Pentland, "Coupled hidden Markov models for complex action recognition," Proc. CVPR, pp.994–999, 1997.

[4] S. Matsuda, M. Nakai, H. Shimodaira, and S. Sagayama, "Asynchronous-transition HMM," Proc. ICASSP, vol. 2, pp. II1005–II1008, 2000.

[5] T. Artières, N. Gauthier, P. Gallinari, and B. Dorizzi, "A hidden Markov model combination framework for handwriting recognition," Int. J. Doc. Anal. Recog., vol. 5, no. 4, pp. 233-243, 2002.

[6] `http://archive.ics.uci.edu/ml/datasets/Pen-Based+Recognition+of+Handwritten+Digits`

[7] T. Flash and N. Hogan, "The coordination of arm movements: an experimentally confirmed mathematical model," Journal of Neuroscience, vol. 5, pp. 1688-1703, 1985.

[8] R. Eddy, A. Krogh, G. Mitchison, R. Durbin, Biological Sequence Analysis, Cambridge University Press, 1999.