A Generative Model for Handwritings Based on Enhanced Feature Desynchronization

Seiichi Uchida, Toru Sasaki, and Feng Yaokai
Kyushu University, Fukuoka, 819-0395, Japan

***Abstract*—A new generative model of handwriting patterns is proposed for interpreting their deformations. The model is based on feature desynchronization, which is a coupling process of x and y coordinate features of different timings. By changing the timings to be coupled, the model can generate various deformed patterns from a single pattern. The model is further enhanced by incorporating an adaptive rotation at each timing for increasing the variety of deformed patterns. An important fact is that this enhanced desynchronization model can be interpreted intuitively as a deformation process in actual handwriting. Experimental results showed that the model can generate various handwriting patterns close to actual deformed patterns.**

***Keywords*-feature desynchronization, handwritings, generative model**

I. Introduction

The purpose of this paper is to introduce a new generative model for interpreting deformations in handwritings. That is, the proposed model imitates a deformation process of human handwriting. In fact, the model gives a simple explanation why we draw a deformed line even if we want to draw just a straight line. In addition, the model can be embedded into elastic matching techniques for realizing deformation-tolerant online character recognition in future.

The proposed model is inspired by *asynchronous* pen movement on drawing a line. For drawing a (perfectly) straight line, a pen should move synchronously in x-direction and y-direction, as shown in Fig. 1 (a). In practice, however, it is impossible to draw a perfectly straight line; we generally have a curved (i.e., deformed) line. A possible interpretation of this unexpected deformation is that the pen usually moves asynchronously in x-direction and y-direction, as shown in Fig. 1 (b).

This asynchronous pen movement can be modeled by *feature desynchronization* [1]. Again, consider a situation of drawing a perfectly straight line from $(0,0)^T$ to $(10,10)^T$. In a discretized manner, the straight line can be represented as a sequence of pen-tip locations, $(0,0)^T, (1,1)^T, (2,2)^T, \ldots, (10,10)^T$. Feature desynchronization is the process of coupling x and y features from *different* timings and thus generates a new point. For example, a new point $(2,3)^T$ is generated from the original points, $(2,2)^T$ and $(3,3)^T$. One important thing is that we can consider that this new point is generated by an asynchronous pen movement where the pen movement in y-direction is a bit faster than that in x-direction. The other important thing is that this new point no longer lies on the original straight line; this means that we can generate various deformations of handwritings by feature desynchronization.

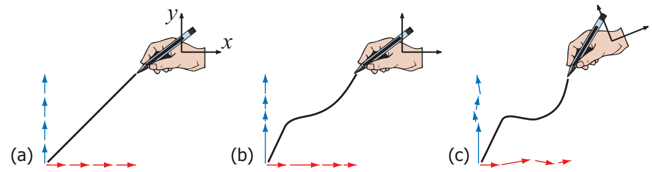

Figure 1. Feature desynchronization for generating deformed handwritings. (a) No desynchronization. (b) Simple desynchronization. (c) Enhanced desynchronization.

This simple feature desynchronization [1] has a limitation in its ability to generate variable deformation patterns. In fact, as discussed later, no deformed pattern is generated from a pattern comprised of line segments parallel to the x- or y-axis. This fact indicates that the ability of generating patterns by the simple feature desynchronization heavily depends on the writing direction and thus the model is not reasonable to be interpreted as a deformation model of human handwriting.

This paper proposes *enhanced feature desynchronization* for a new generative model with more reasonable ability to generate deformed patterns. In the proposed model, an *adaptive rotation* of the $x-y$ coordinate system is newly introduced into the feature desynchronization process. Simply speaking, this adaptive rotation can remove the above limitation because the $x-y$ coordinate system is no longer fixed.

A possible application of the proposed model is deformation-tolerant online character recognition. Specifically, the model can be embedded into some elastic matching method, such as the dynamic programming (DP) matching method [2]–[4]. By nonlinear timing alignment by elastic matching and adaptive deformation by the enhanced feature desynchronization, it is able to fit a reference pattern to an input pattern closely. This ability will be experimentally shown in this paper.

It should be emphasized again that the proposed model can be considered as a new generation model based on a deformation process of human handwritings. In this point, the proposed method is related to well-known handwriting models, such as Flash and Hogan [5] and the log-normal model [6]. However, the proposed model is far different from those models, which focus dynamics or kinematics of the handwriting process. In contrast, the proposed model focuses

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.124

IEEE computer society

structural fluctuation, i.e., deformations, in a direct manner.

The idea of feature desynchronization is rarely found in literatures, but several trials in other recognition tasks [7]–[9]. Artières et al. [10] have coupled online and offline features for handwriting recognition. Their trial does not aim to model deformations in handwritings and therefore is very different from ours.

II. Generative Model Based on Simple Feature Desynchronization

A. Simple Feature Desynchronization [1]

Consider feature desynchronization of a handwriting pattern $\boldsymbol{R} = \boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_j, \ldots, \boldsymbol{r}_J$, where $\boldsymbol{r}_j$ is the x-y coordinate feature vector of the timing j, i.e., $\boldsymbol{r}_j = (x_j, y_j)^T$. As noted in Section I, feature desynchronization is a process of coupling x and y features of different timings. As shown in Fig. 2 (a), a new point $\widetilde{\boldsymbol{r}} = (x_l, y_k)^T$ is generated by the coupling. This fact indicates that feature desynchronization over the whole pattern will generate a new pattern.

A generated pattern with I couples is represented as $\widetilde{\boldsymbol{R}} = \widetilde{\boldsymbol{r}}_1, \ldots, \widetilde{\boldsymbol{r}}_i, \ldots, \widetilde{\boldsymbol{r}}_I$, where $\widetilde{\boldsymbol{r}}_i = (x_{j(i)}, y_{k(i)})^T$. The functions $j = j(i)$ and $k = k(i)$ specify feature desynchronization. By controlling those functions, it is possible to generate various patterns from a single pattern $\boldsymbol{R}$.

For regulating the effect of desynchronization, we assume the following three constraints.

- The first constraint is monotonicity and continuity conditions for preserving the original temporal order in the both of x and y feature sequences and avoiding large temporal jumps:
$$\begin{cases} j(i) - j(i-1) \in \{0, 1, 2\}, \\ k(i) - k(i-1) \in \{0, 1, 2\}. \end{cases} \qquad (1)$$
- The second constraint is a *constant lag limitation* to avoid coupling x_j and y_k of very different timings:
$$|k(i) - j(i)| \leq L, \qquad (2)$$
where L is a positive constant. A simple extension of this constraint is an *adaptive lag limitation*, where the different values $L(i)$ are used at different i instead of the constant L.
- The third constraint is the boundary condition that at least one of $j(1)$ and $k(1)$ equals to 1 and, similarly, at least one of $j(I)$ and $k(I)$ equals to J. (Note that this boundary condition can be modified to be more strict or more loose. For example, a strict version is the condition that $j(1) = k(1) = 1$ and $j(I) = k(I) = J$.)

B. Limitation of Simple Feature Desynchronization

The above simple feature desynchronization has a serious limitation that the variety of generated patterns depends on the stroke direction. Consider an extreme case that the handwriting pattern is just a vertical line. In this case, all of the x-coordinate features (i.e., $x_1, \ldots, x_J$) have the same value and any feature coupling cannot make a new point. This is because a generated point (x_j, y_k) is equivalent to an original point (x_k, y_k) and thus no deformed pattern can be generated from the vertical line by feature desynchronization.

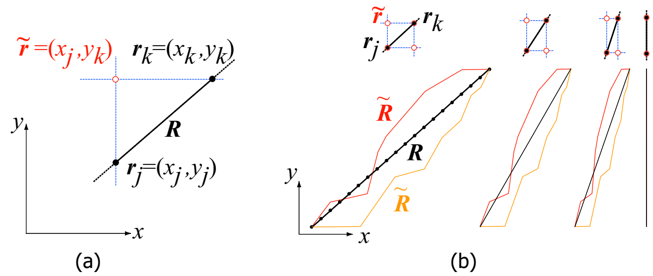

Figure 2. Simple feature desynchronization [1]. (a) A newly generated point. (b) Limitation in the variety of generated patterns.

This limitation appears even in more general cases. Figure 2 (b) illustrates the dependency between the stroke direction and the variety of generated patterns. If the stroke direction becomes closer to 0°(or 90°), the range of the x (or y) feature values is limited. Since the patterns are generated by using the original feature values, the variety of the generated patterns is limited for the patterns with those directions. Although Fig. 2 (b) uses a straight line as the original pattern $\boldsymbol{R}$ for simplicity, any other complex pattern also has the same limitation depending on its local directions.

III. Generative Model Based on Enhanced Feature Desynchronization

In this section, an enhanced feature desynchronization is introduced for building a new generative model which is free from the above limitation. The main idea of the enhanced feature desynchronization is an adaptive rotation of the x-y coordinate system. Figure 3 (a) shows the generation of a new point (x_j^θ, y_k^θ) under a rotation θ. Comparison to Fig. 2 (a) shows that a different point can be generated by the rotation.

The generated point (x_j^θ, y_k^θ) is described as

$$\begin{cases} x_j^\theta &= (y_j - y_k)\sin\theta\cos\theta + x_j\cos^2\theta + x_k\sin^2\theta, \\ y_k^\theta &= (x_j - x_k)\sin\theta\cos\theta + y_j\sin^2\theta + y_k\cos^2\theta. \end{cases} \qquad (3)$$

By eliminating θ from these equations, we have

$$\begin{aligned} (x_j^\theta - (x_j + x_k)/2)^2 &+ (y_k^\theta - (y_j + y_k)/2)^2 \\ &= \left((x_j - x_k)^2 + (y_j - y_k)^2\right)/4. \end{aligned} \qquad (4)$$

This indicates that the new point $\widetilde{\boldsymbol{r}}^\theta = (x_j^\theta, y_k^\theta)$ by the enhanced feature desynchronization lies on the circle whose diameter is defined by (x_j, y_j) and (x_k, y_k). Figure 3 (b) illustrates the circle. Note that this circle does not depend on the center of the rotation.

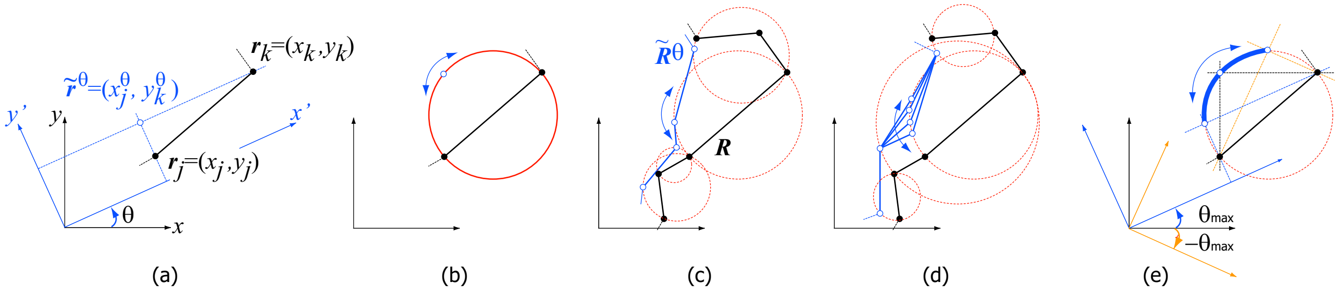

Figure 3. Enhanced feature desynchronization. (a) A point generated by enhanced feature desynchronization. (b) The circle representing all points to be generated. (c) and (d) Two examples of $\widetilde{\boldsymbol{R}}^\theta$ from the same $\boldsymbol{R}$. (e) Effect of rotation angle limitation.

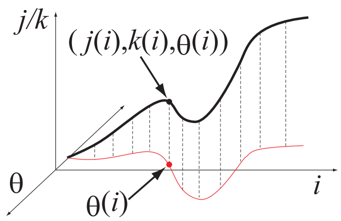

Figure 4. Optimization of $j(i)$, $k(i)$, and $\theta(i)$ by DP.

In the enhanced desynchronization, the rotation angle θ is changed adaptively at every i. Hereafter, $\theta(i)$ denotes the rotation angle at i and $\widetilde{\boldsymbol{R}}^\theta$ denotes a generated pattern. It should be noted that $\theta(i)$ simulates a fluctuation of hand posture as shown in Fig. 1 (c). While we draw a stroke, our hand posture (especially, wrist angle) varies. Since the deformation characteristics will depend on the hand posture, it is natural to introduce $\theta(i)$ into the model.

Figure 3 (c) and (d) illustrate two examples of generated patterns $\widetilde{\boldsymbol{R}}^\theta$ from a single pattern $\boldsymbol{R}$. Since there are many possible couplings of two points (even under a lag limitation), many different patterns can be generated from the same pattern. Furthermore, even from the same point couplings, many patterns can be generated by moving points along circles, that is, by changing $\theta(i)$, as shown in (d).

According to the introduction of the adaptive rotation, we must employ two more constraints in addition to the three constraints of Section II-A.

- The fourth constraint is the rotation angle limitation to limit $\theta(i) \in [-\theta_{\max}, \theta_{\max}]$ shown in Fig. 3 (e). Under this constraint, $\widetilde{\boldsymbol{r}}_i$ is forced to lie on a certain arc instead of the whole circle.
- The fifth constraint is the continuity of $\theta(i)$. From the analogy of Fig 1 (c), the rotation angle $\theta(i)$ should not change abruptly and thus we employ the continuity constraint $|\theta(i) - \theta(i-1)| \leq \theta_{\max}/2$.

IV. Embedding the Model into Elastic Matching

The proposed generative model has a potential of realizing deformation-tolerant online character recognition. Assume that $\boldsymbol{R}$ is a reference pattern of a certain class, and $\boldsymbol{E} = \boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_i, \ldots, \boldsymbol{e}_I$ is an input pattern to be recognized. The input pattern $\boldsymbol{E}$ will be correctly recognized regardless its deformation, if we find $\widetilde{\boldsymbol{R}}^\theta \sim \boldsymbol{E}$.

The remaining problem is how to generate the optimal $\widetilde{\boldsymbol{R}}^\theta$ for given $\boldsymbol{E}$ and $\boldsymbol{R}$. In other words, we must find $\widetilde{\boldsymbol{R}}^\theta$ which is closet to $\boldsymbol{E}$ among all possible $\widetilde{\boldsymbol{R}}^\theta$s. One possible way is to formulate the minimization problem of the following objective function $\mathcal{J}$ with respect to $\{j(i), k(i), \theta(i) | i = 1, \ldots, I\}$ subject to the five constraints of Section II-A and Section III:

$$\mathcal{J} = \sum_{i=1}^{I} \|\boldsymbol{e}_i - \widetilde{\boldsymbol{r}}_i^\theta\|, \tag{5}$$

where $\widetilde{\boldsymbol{r}}_i^\theta = \left(x_{j(i)}^{\theta(i)}, y_{k(i)}^{\theta(i)}\right)^T$.

The optimization problem is an extended version of the conventional elastic matching problem and thus solved efficiently by dynamic programming (DP). Letting $d_i(j,k,\theta) = \|\boldsymbol{e}_i - \widetilde{\boldsymbol{r}}_i^\theta\|$, the DP algorithm is organized as calculation of the following recursive equation at all (j,k,θ) from $i = 1$ to I:

$$g_i(j,k,\theta) = d_i(j,k,\theta) + \min_{j',k',\theta'} g_{i-1}(j',k',\theta'), \tag{6}$$

where (j',k',θ') should satisfies all the constraints with (j,k,θ). For example, $j - j' \in \{0,1,2\}$ and $|j-k| \leq L$. Consequently, $\min_{j,k,\theta} g_I(j,k,\theta)$ is the minimum matching cost between $\boldsymbol{E}$ and $\boldsymbol{R}$ and the optimal $\{j(i), k(i), \theta(i)\}$ is given as a path of Fig. 4. The class of the reference pattern giving the minimum matching cost becomes the recognition result of $\boldsymbol{E}$.

V. Experimental Results

A. Qualitative Evaluation

An experiment has conducted to observe the ability of the proposed model on generating deformation patterns. Specifically, the DP-based elastic matching with the proposed model was performed between a pair of digit patterns $\boldsymbol{R}$ and $\boldsymbol{E}$ from the same class. If $\widetilde{\boldsymbol{R}}^\theta$ is close to $\boldsymbol{E}$, it proves that the proposed model can generate patterns close to actual deformed patterns (i.e., patterns by human handwriting) in

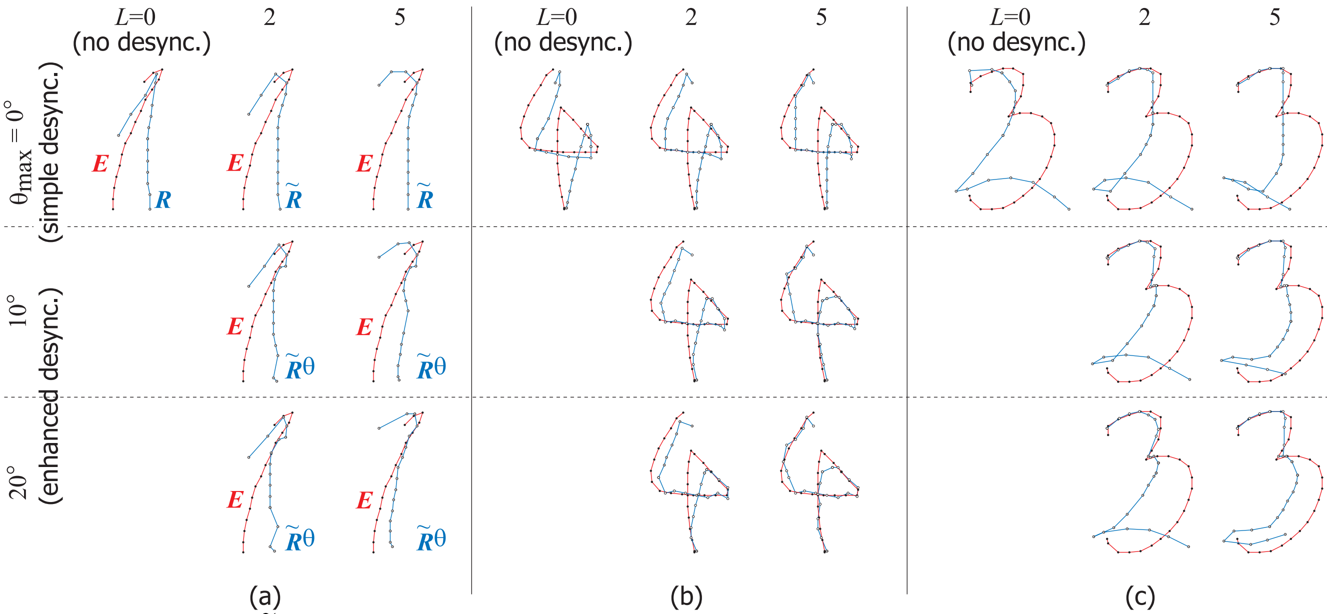

Figure 5. Generated pattern $\widetilde{\boldsymbol{R}}^{\theta}$ (blue) optimally fitted the input pattern $\boldsymbol{E}$ (red) under different values of the maximum lag L and the maximum rotation angle $\theta_{\max}$. The original pattern $\boldsymbol{R}$ is shown at $L = 0$.

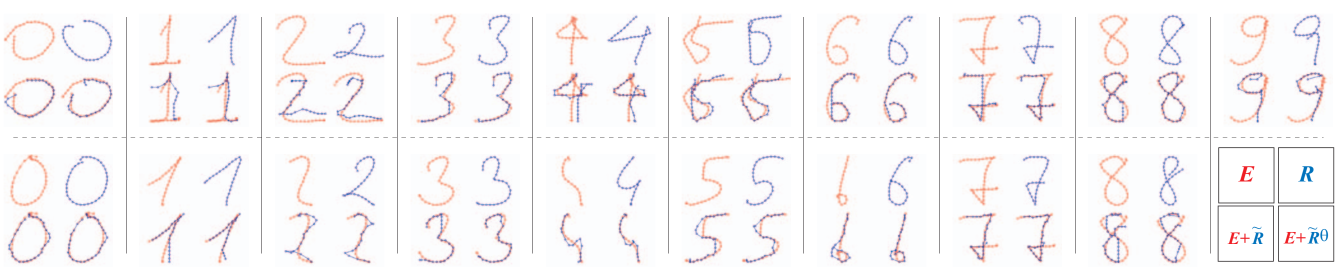

Figure 6. Patterns $\boldsymbol{E}$ and $\boldsymbol{R}$, and generated patterns $\widetilde{\boldsymbol{R}}$ and $\widetilde{\boldsymbol{R}}^{\theta}$.

the same class. Note that the public online digit database called Ethem Alpaydin Digit was used for the experiment.

Figure 5 shows the results. The patterns under $L = 0$ are original patterns $\boldsymbol{R}$ and $\boldsymbol{E}$. Note that conventional elastic matching methods, such as the DP-based elastic matching, evaluates the geometric difference of those two original patterns under the optimal point correspondence.

First, let us observe $\widetilde{\boldsymbol{R}}$ in Fig. 5, which are results by elastic matching with the simple feature desynchronization (or, equivalently, the enhanced model under $L \neq 0$ and $\theta_{\max} = 0$). A pair of "1" of Fig. 5 (a) clearly shows the limit of the simple feature desynchronization because the near-vertical stroke of "1" cannot change its direction largely and there is still a significant difference between $\widetilde{\boldsymbol{R}}$ and $\boldsymbol{E}$. A pair of "4" in Fig. 5 (b) shows that the ending part of "4", which is originally slanted and thus not parallel to the x- or y-axis, was deformed to be fitted to $\boldsymbol{E}$ with the simple model. A larger L provided a better fitting because it will increase pattern generation ability. In contrast, it is possible to observe a clear limitation of the simple model at the beginning part of "4"; simply speaking, no point can be generated outside of the bounding box of an original stroke and thus the beginning part is unnaturally deformed like a vertical stroke.

Second, let us observe $\widetilde{\boldsymbol{R}}^{\theta}$ in Fig. 5, which are results elastic matching with the enhanced feature desynchronization. It is obvious that the pattern generation ability is increased by the introduction of the adaptive rotation $\theta(i)$. The pattern $\widetilde{\boldsymbol{R}}^{\theta}$ of "1" no longer remains as a vertical stroke. The pattern $\widetilde{\boldsymbol{R}}^{\theta}$ of "4" shows a close fit to $\boldsymbol{E}$ (even around its beginning part) especially under $L = 5$ and $\theta_{\max} = 10$. These two results of Fig. 5 (a) and (b) prove the enhanced model can generate patterns $\widetilde{\boldsymbol{R}}^{\theta}$ close to actual deformed patterns $\boldsymbol{E}$.

As expected, an *overfitting* phenomenon happened between patterns from different classes. Figure 5 (c) shows an example of the overfitting, where a reference pattern $\boldsymbol{R}$ of "2" is fitted to $\boldsymbol{E}$ of "3" by using the higher pattern generation ability of the proposed model.

One possible remedy to eliminate the overfitting in future is to *learn* the optimal constraint parameters using training patterns. For example, the adaptive lag $L(i)$ can be designed for each class. In [1], a trial of training $L(i)$ have been made

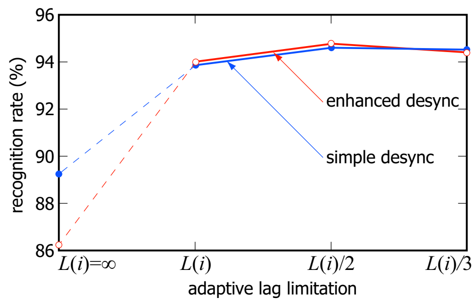

Figure 7. Recognition rates.

on the simple desynchronization model and an improvement on eliminating the overfitting has been reported. The proposed model has another parameter $\theta(i)$ to be learned. One interesting future work is to investigate whether $\theta(i)$ is class-dependent or writer-dependent. The latter comes from the analogy between the adaptive rotation and the hand posture as indicated in Fig. 1 (c).

Figure 6 shows other results between patterns of the same class. In all results, $\widetilde{\boldsymbol{R}}^{\theta}$ shows a closer fitting than $\widetilde{\boldsymbol{R}}$. In several cases (e.g., the upper examples of "6" and "8"), $\widetilde{\boldsymbol{R}}^{\theta}$ shows almost perfect fitting to $\boldsymbol{E}$. Again, it should be emphasized that those close fitting is realized just by feature desynchronization and rotation.

B. Quantitative Evaluation

An online character recognition experiment has been conducted, although the main purpose of this paper is to propose a new generative model for interpreting deformation of human handwriting and observe its deformation ability. For each of 10 digit classes, 10 reference patterns were selected from the training set of the database by k-means. For the rotation angle limitation, $\theta_{\max}$ was fixed at $20°$. The adaptive lag limitation was also used. Its parameter $L(i)(i = 1, \ldots, I)$ was determined for each reference pattern as the maximum lag observed in desynchronization matching results between training 300 pattern pairs from the same category.

Figure 7 shows the recognition rates by the DP-based elastic matching with the simple and the enhanced feature desynchronization models. As expected, without the lag limitation (i.e., $L(i) = \infty$), the enhanced model shows a poor performance because of the overfitting problem. The performance was improved by limiting the lag by $L(i)$. Under more severe lag limitation by $L(i)/2$, the performance by the enhanced model was further improved and slightly better than the performance by the simple model. This indicates that the adaptive rotation does not cause severe overfitting if we set appropriate constraints. In this case, 28 samples turned to be recognized correctly by the enhanced model, and 22 samples turned to be misrecognized.

VI. Conclusion

A new generative model based on enhanced feature desynchronization has been proposed for modeling the deformations of human handwritings. An important point is that the proposed model is based on a simple structural and combinatorial framework. In spite of this simplicity, the model can generate patterns close to actual deformed patterns just from a single pattern. This fact has been proved via several experimental results and, consequently, it has also been proved that the enhanced feature desynchronization can be treated as a handwriting model.

Future work will focus mainly on the following points.

- Class-dependent and writer-dependent characteristics should be incorporated into the model. This can be realized by learning the constraint parameters (e.g., $L(i)$) via some training steps.
- The variations in the resulting adaptive rotation angle sequences $\theta(1), \ldots, \theta(i), \ldots, \theta(I)$ should be observed within a class or within a writer. Then, the relation between the rotation angle and the hand posture should be clarified.
- The enhanced desynchronization model should be further extended toward the realization of accurate and deformation-tolerant online character recognition systems. Again, the proposed desynchronization model is simple and thus there are various ways for the extension.

Acknowledgment

This research was partially supported by JST, CREST.

References

[1] S. Uchida, K. Niyagawa, and H. Sakoe, "Feature desynchronization in online character recognition," ICFHR, 2008.

[2] K. Yoshida and H. Sakoe, "Online handwritten character recognition for a personal computer system," IEEE Trans. CE, 28(3), 1982.

[3] S. D. Connell and A. K. Jain, "Template-based online character recognition," PR, 34(1), 2001.

[4] C. Bahlmann and H. Burkhardt, "The writer independent online handwriting recognition system *frog on hand* and cluster generative statistical dynamic time warping," PAMI, 26(3), 2004.

[5] T. Flash and N. Hogan, "The coordination of arm movements: an experimentally confirmed mathematical model," J. Neuroscience, 5, 1985.

[6] R. Plamondon, "Neuromuscular studies of handwriting generation and representation," ICFHR, (Invited Talk), 2010.

[7] M. Brand, N. Oliver, and A. Pentland, "Coupled hidden Markov models for complex action recognition," CVPR, 1997.

[8] S. Matsuda, M. Nakai, H. Shimodaira, and S. Sagayama, "Asynchronous-transition HMM," ICASSP, 2000.

[9] A. V. Nefian, L. Liang, X. Pi, L. Xiaoxiang, C. Mao and K. Murphy, "A coupled HMM for audio-visual speech recognition," ICASSP, 2002.

[10] T. Artières, N. Gauthier, P. Gallinari, and B. Dorizzi, "A hidden Markov model combination framework for handwriting recognition," IJDAR, 5(4), 2002.