PREWARPING SIAMESE NETWORK: LEARNING LOCAL REPRESENTATIONS FOR ONLINE SIGNATURE VERIFICATION

Xiaomeng Wu[†], *Akisato Kimura*[†], *Seiichi Uchida*[‡], *and Kunio Kashino*[†]

[†] NTT Communication Science Laboratories, NTT Corporation, Kanagawa, Japan
[‡] Department of Advanced Information Technology, Kyushu University, Fukuoka, Japan

ABSTRACT

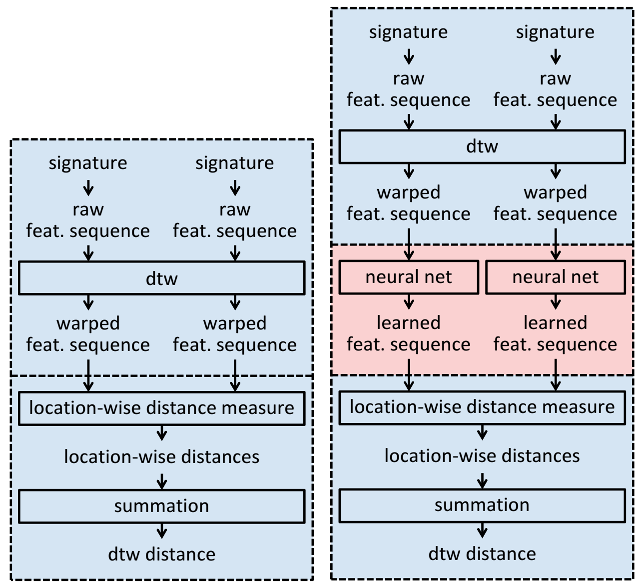
We propose a neural network-based framework for learning local representations of multivariate time series, and demonstrate its effectiveness for online signature verification. In contrast to related works that optimize a global distance objective, we incorporate a Siamese network into dynamic time warping (DTW), leading to a novel prewarping Siamese network (PSN) optimized with a local embedding loss. PSN learns a feature space that preserves the temporal location-wise distances of local structures. Local embedding, along with the alignment conditions of DTW, imposes a temporal consistency constraint on the sequence-level distance measure while achieving invariance as regards non-linear distortions. Validation on online signature verification datasets demonstrates the advantage of our framework over existing techniques that use either handcrafted or learned feature representations.

Index Terms — Biometrics, DTW, feature learning, online signature verification, Siamese network

1. INTRODUCTION

Biometrics is deployed widely in a multitude of applications, e.g., security, e-government, health care, education, banking, and insurance [1]. Among the wide range of biometric traits, handwritten signatures have long been established as the most widespread means of personal verification [2]. In this paper, we focus on online signature verification, where the signature is represented as a multivariate time series sampled with an online acquisition device during the writing process. The task is to evaluate the authenticity of a test signature by matching it against a few reference specimens enrolled in a database. It remains challenging because of large intra-personal variability and subtle differences between genuine and forged signatures. One of the most critical aspects of this task is signature matching, i.e., how to define a discriminative distance measure between two time series.

Dynamic time warping (DTW) [3] has been extensively used for this purpose [4–12]. It calculates an optimal match between two time series, which has the minimal cost computed as the sum of the distances, for each matched pair of temporal locations, between their local features. Defining a distance with this minimal cost, DTW achieves invariance to non-linear distortions and also imposes a temporal consistency constraint on verification. Meanwhile, intensive research has been devoted to HMMs [12–16], which have been found to be well suited for signature modeling since they are highly adaptable to personal variability. In earlier work [17], a Siamese network was used to learn a distance metric from data, driving the distance to be small for pairs of genuine signatures from the same subject and longer for a pair consisting of a genuine and a forged signature. Most of such Siamese networks [17–21] produce a global representation for the time series by flattening [17], average pooling [20], or using



(a) DTW (b) Proposed method

Fig. 1. Incorporating Siamese network into dynamic time warping.

fully connected layers [19, 21] on hidden unit activations. However, the loss of temporal information makes such a global representation incompatible with temporal consistency-aware models as involved in DTW.

In this paper, we make it a goal to learn neural network-based local representations for online signature verification. The learned network should be able to impose a temporal consistency constraint on the distance measure between two time series while achieving invariance when encountering non-linear distortions, e.g., temporal translations and scaling. To this end, we revisit DTW and decompose it into warping and distance computation processing steps. A Siamese network is then incorporated between the two steps as shown in Fig. 1, leading to a novel prewarping Siamese network (PSN). A PSN inherits the temporal invariance from DTW and so is highly adaptable to intra-personal variability (robustness). Unlike related work producing global representations [17–21], PSN accurately learns a metric space that captures temporal location-wise distances of local structures. These two properties together impose a temporal consistency constraint on the verification: two temporally aligned time series are determined as a matching pair only if their local representations at the same location are sufficiently similar to each other (discriminative power).

2. PROPOSED METHOD

Let $\mathcal{X}$ denote a test signature. Let $\{\mathcal{Y}^1, \ldots, \mathcal{Y}^n\}$ be a set of n reference specimens that are enrolled in a database corresponding to a claimed identity. The task is to evaluate the authenticity of $\mathcal{X}$ by matching it against $\{\mathcal{Y}^1, \ldots, \mathcal{Y}^n\}$. Given a distance measure $d(\cdot, \cdot)$ between two time series, we define the dissimilarity between $\mathcal{X}$ and the specimens using Eq. 1. The final decision on signature authenticity can then be achieved with a subject-independent threshold classifier. The dissimilarity can also be normalized as in Eq. 2 by the mean value of all the pairwise distances within $\{\mathcal{Y}^1, \ldots, \mathcal{Y}^n\}$ for greater robustness as regards inter-device and interpersonal variability. The problem is how to define and learn the distance measure $d(\cdot, \cdot)$.

$$D(\mathcal{X}, \{\mathcal{Y}^1, \ldots, \mathcal{Y}^n\}) = \sum_{i=1}^{n} d(\mathcal{X}, \mathcal{Y}^i) \tag{1}$$

$$D(\mathcal{X}, \{\mathcal{Y}^1, \ldots, \mathcal{Y}^n\}) = \frac{\sum_{i=1}^{n} d(\mathcal{X}, \mathcal{Y}^i)}{\sum_{i=1}^{n}\sum_{j>i}^{n} d(\mathcal{Y}^i, \mathcal{Y}^j)} \tag{2}$$

2.1. Dynamic Time Warping

We first revisit the DTW algorithm. Let $\mathcal{X} = [\mathcal{X}_1 \ldots \mathcal{X}_N]^\top$ where $\mathcal{X}_i$ is a column vector of raw, handcrafted local features at temporal location i in $\mathcal{X}$. The warping path $\mathbf{\Pi}$ between two time series $\mathcal{X}$ and $\mathcal{Y}$ of lengths N and M, respectively, is a pair of increasing integral vectors (π_1, π_2) of length $L < N+M$ such that $1 = \pi_1(1) \leq \cdots \leq \pi_1(L) = N$ and $1 = \pi_2(1) \leq \cdots \leq \pi_2(L) = M$ (monotonicity), with unitary increments (continuity). In its typical form, the optimal warping path $\hat{\mathbf{\Pi}}$ is defined as

$$\hat{\mathbf{\Pi}} = \arg\min_{\mathbf{\Pi}} \sum_{i=1}^{L} \|\mathcal{X}_{\pi_1(i)} - \mathcal{Y}_{\pi_2(i)}\|. \tag{3}$$

With a $\hat{\mathbf{\Pi}}$ of length $\hat{L}$, $\mathcal{X}$ and $\mathcal{Y}$ can be aligned in time, i.e., transformed to two warped feature sequences $\hat{\mathcal{X}}$ and $\hat{\mathcal{Y}}$ of the same length $\hat{L}$ such that

$$\hat{\mathcal{X}} = [\mathcal{X}_{\hat{\pi}_1(1)} \ldots \mathcal{X}_{\hat{\pi}_1(\hat{L})}]^\top,\ \hat{\mathcal{Y}} = [\mathcal{Y}_{\hat{\pi}_2(1)} \ldots \mathcal{Y}_{\hat{\pi}_2(\hat{L})}]^\top. \tag{4}$$

In online signature verification based on DTW, the minimal cost in Eq. 3 defines the distance measure $d(\mathcal{X}, \mathcal{Y})$. It actually equals the sum of location-wise distances between $\hat{\mathcal{X}}$ and $\hat{\mathcal{Y}}$ as in Eq. 5, where $\hat{\mathcal{X}}_i$ and $\hat{\mathcal{Y}}_i$ are local feature vectors at location i.

$$d(\mathcal{X}, \mathcal{Y}) = \sum_{i=1}^{\hat{L}} \|\hat{\mathcal{X}}_i - \hat{\mathcal{Y}}_i\| \tag{5}$$

From this viewpoint, we decompose DTW into two processing steps: 1) finding the optimal warping path $\hat{\mathbf{\Pi}}$ and transforming $\mathcal{X}$ and $\mathcal{Y}$ to $\hat{\mathcal{X}}$ and $\hat{\mathcal{Y}}$ as in Eqs. 3 and 4; 2) computing the sum of the location-wise distances as in Eq. 5. These two steps are shown as two blocks in Fig. 1a.

2.2. Prewarping Siamese Network

To allow feature learning, we propose incorporating a Siamese network between the two processing steps of DTW as in Fig. 1b. This network takes warped feature sequences $\hat{\mathcal{X}}$ and $\hat{\mathcal{Y}}$ as inputs and produces two sequences of hidden unit activations as outputs. These activations should have a natural interpretation as features of local signature segments, rather than global representations, corresponding to the receptive fields of each feature vector. Typical examples of such a network include fully convolutional networks and bidirectional RNNs. Let $\mathbf{X}$ and $\mathbf{Y}$ of the same length W denote the feature sequences output from the Siamese network. We replace Eq. 5 with an alternative distance defined by using the location-wise distances between $\mathbf{X}$ and $\mathbf{Y}$ as in Eq. 6, where $\mathbf{x}_i$ and $\mathbf{y}_i$ are feature vectors at location i.

$$d(\mathcal{X}, \mathcal{Y}) = \frac{1}{W} \sum_{i=1}^{W} \|\mathbf{x}_i - \mathbf{y}_i\| \tag{6}$$

In this way, DTW makes Eq. 6 invariant to temporal distortions, and improves the accuracy of the local correspondences $\mathbf{x}_i$ and $\mathbf{y}_i$. Meanwhile, Eq. 6, together with the monotonicity and continuity conditions described in Section 2.1, imposes a temporal consistency constraint on the distance measure: two temporally aligned time series are determined as matching only if their local segments at the same location are similar to each other. Our method thus avoids the loss of temporal information compared with existing Siamese networks that learn global signature representations.

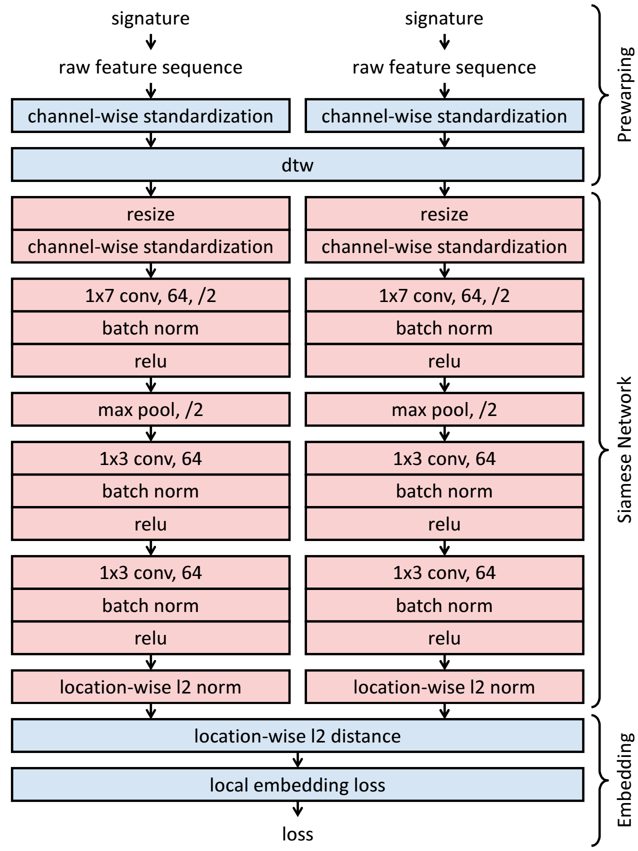


Fig. 2. Prewarping Siamese network.

Figure 2 shows the PSN architecture in detail. Given an input signature, its pen coordinates are used to extract 7 of 27 time functions at each temporal location [16], resulting in a sequence of 7D local features. After channel-wise standardization, the standardized sequences of two inputs are warped and aligned in time with DTW.

To enable PSN to work on mini-batches, the warped feature sequences are resized to a predefined length (1,024 in this paper) followed by another standardization. The sub-networks of PSN can be any neural network as long as its hidden unit activations can be interpreted as a sequence of local features with sufficiently high temporal resolution. In this paper, we use a fully convolutional network as the sub-network because of its advantages in terms of parallelism and stable gradient computation over RNNs. The input sequence of this network can be understood as a 7-channel "image" of size 1×1024, the width of which actually corresponds to the temporal length. This one-dimensional CNN contains one 1×7 convolution, one max pooling, and two 1×3 convolution layers. The batch normalization [22] is used right after each convolution. The last ReLU layer is followed by location-wise l^2-normalization. The feature sequences $\mathbf{X}$ and $\mathbf{Y}$ output from the sub-networks of PSN are thus two $1\times W\times K$ tensors or two $W\times K$ matrices, where $W = 256$ is the temporal length and $K = 64$ the number of dimensions.

A PSN is trained by minimizing the local embedding loss as in Section 2.3. During testing, the distance between two signatures is computed by Eq. 6 and substituted into Eq. 1 or 2 for verification.

2.3. Local Embedding Loss

Learning the distance between two corresponding time series causes their temporal structures to be mapped to two neighboring points in a learned feature space. In a typical Siamese network [20, 23], this is usually done with a pooling operation, e.g., average pooling, on the activations of the last layer to obtain a global representation. The Siamese network is trained by minimizing a contrastive loss [24]. Let $\bar{\mathbf{x}} = [\bar{x}_1 \dots \bar{x}_K]^\top$ be a global representation constructed with average pooling, where $\bar{x}_j = \frac{1}{W}\sum_{i=1}^{W} x_{ij}$. Here, x_{ij} is the activation at location i and dimension j in $\mathbf{X} \in \mathbb{R}^{W\times K}$. Let $z \in \{0, 1\}$ declaring whether a pair is non-matching or matching, respectively. The global contrastive loss and its derivative can be defined by Eqs. 7 and 8 with τ being the margin of the hinge loss.

$$\mathcal{L} = \begin{cases} \frac{1}{2}\|\bar{\mathbf{x}} - \bar{\mathbf{y}}\|^2 & \text{if } z = 1 \\ \frac{1}{2}[\max(0, \tau - \|\bar{\mathbf{x}} - \bar{\mathbf{y}}\|)]^2 & \text{otherwise} \end{cases} \tag{7}$$

$$\frac{\partial\mathcal{L}}{\partial x_{ij}} = \begin{cases} \frac{\bar{x}_j - \bar{y}_j}{W} & \text{if } z = 1 \\ -\frac{\bar{x}_j - \bar{y}_j}{W} \cdot \frac{\max(0, \tau - \|\bar{\mathbf{x}} - \bar{\mathbf{y}}\|)}{\|\bar{\mathbf{x}} - \bar{\mathbf{y}}\|} & \text{otherwise} \end{cases} \tag{8}$$

However, this global loss usually leads to the disappearance of temporal information. From Eq. 8, we can see that the gradient at the activation x_{ij} only relies on $\bar{x}_j - \bar{y}_j$ and $\|\bar{\mathbf{x}} - \bar{\mathbf{y}}\|$, which are totally independent of the location i. That is, all x_{ij} at dimension j receive the same updates at each training iteration. In this paper, we propose a generalization of the contrastive loss, called local embedding loss, to learn local representations from corresponding local segments, instead of the whole time series, of two signatures.

Location-wise local embedding loss. One method of achieving this purpose is to define a contrastive loss for each temporal location and then to aggregate the losses from all locations. Recall that $\mathbf{x}_i \in \mathbf{X}$ and $\mathbf{y}_i \in \mathbf{Y}$ are feature vectors at location i with $i \in \{1, \dots, W\}$, and that $x_{ij} \in \mathbf{x}_i$ is the activation at location i and dimension j. Our location-wise local embedding loss and its derivative with respect to x_{ij} can thus be defined by Eqs. 9 and 10.

$$\mathcal{L} = \begin{cases} \frac{1}{2W}\sum_{i=1}^{W}\|\mathbf{x}_i - \mathbf{y}_i\|^2 & \text{if } z = 1 \\ \frac{1}{2W}\sum_{i=1}^{W}[\max(0, \tau - \|\mathbf{x}_i - \mathbf{y}_i\|)]^2 & \text{otherwise} \end{cases} \tag{9}$$

$$\frac{\partial\mathcal{L}}{\partial x_{ij}} = \begin{cases} \frac{x_{ij} - y_{ij}}{W} & \text{if } z = 1 \\ -\frac{x_{ij} - y_{ij}}{W} \cdot \frac{\max(0, \tau - \|\mathbf{x}_i - \mathbf{y}_i\|)}{\|\mathbf{x}_i - \mathbf{y}_i\|} & \text{otherwise} \end{cases} \tag{10}$$

Compared with Eq. 8, the gradient in Eq. 10 relies on $x_{ij} - y_{ij}$ and $\|\mathbf{x}_i - \mathbf{y}_i\|$, both being dependent on the location i. For non-matching pairs, the activations at location i that have a smaller location-wise distance $\|\mathbf{x}_i - \mathbf{y}_i\|$ receive larger parameter updates during the training process.

Sequence-wise local embedding loss. From another viewpoint, we can also define the contrastive loss directly by using the sequence-level distance measure $d(\mathcal{X}, \mathcal{Y})$ in Eq. 6, instead of defining a contrastive loss for each temporal location. Recall that $d(\mathcal{X}, \mathcal{Y})$ is the average of the location-wise l^2 distances between each pair of $\mathbf{x}_i$ and $\mathbf{y}_i$. In consequence, our sequence-wise local embedding loss and its derivative can be defined by Eqs. 11 and 12.

$$\mathcal{L} = \begin{cases} \frac{1}{2}[d(\mathcal{X}, \mathcal{Y})]^2 & \text{if } z = 1 \\ \frac{1}{2}\{\max[0, \tau - d(\mathcal{X}, \mathcal{Y})]\}^2 & \text{otherwise} \end{cases} \tag{11}$$

$$\frac{\partial\mathcal{L}}{\partial x_{ij}} = \begin{cases} \frac{x_{ij} - y_{ij}}{W} \cdot \frac{d(\mathcal{X}, \mathcal{Y})}{\|\mathbf{x}_i - \mathbf{y}_i\|} & \text{if } z = 1 \\ -\frac{x_{ij} - y_{ij}}{W} \cdot \frac{\max[0, \tau - d(\mathcal{X}, \mathcal{Y})]}{\|\mathbf{x}_i - \mathbf{y}_i\|} & \text{otherwise} \end{cases} \tag{12}$$

Similar to Eq. 10, the gradient in Eq. 12 is location-adaptive. The difference between the two local embedding losses lies in their dependence on $d(\mathcal{X}, \mathcal{Y})$. Parameter updates are dampened for matching pairs if they already have a small $d(\mathcal{X}, \mathcal{Y})$. On the other hand, non-matching pairs contribute to the loss function only if their $d(\mathcal{X}, \mathcal{Y})$ is within a radius defined by the margin τ.

3. EXPERIMENTS

3.1. Datasets

In this paper, we considered the most common type of verification: deciding whether a test signature is a genuine signature or a skilled forgery in relation to a claimed identity. Here, a skilled forgery indicates a signature imitated by a forger. We focused on four signature datasets: MCYT-100 [25], BiosecurID SONOF [26], and SUSIG visual and blind sub-corpuses [6]. These four datasets vary widely as regards acquisition protocol, geographical location, and registering device. MCYT-100 is the largest of the four datasets and contains 100 subjects, each having 25 genuine signatures and 25 skilled forgeries. Considering the limited amount of available training data, we validated our method under the following two experimental protocols based on the four datasets.

MCYT-100 (90/80/70%). The first 90/80/70% of subjects in MCYT were used for training, and the remaining subjects for testing.

FULL. We combined all the four datasets: the first 90% of subjects in each dataset were extracted and combined for training; all the remaining subjects were left for testing. The training set contains 375 subjects and 11,944 signatures; the testing set contains 41 subjects and 1,312 signatures. This protocol has a larger amount of training data, but is challenging because of the large inter-device variability.

For each subject in the testing set, the first five genuine signatures were used as reference signatures. The remaining genuine signatures and all the skilled forgeries were used as test signatures. The dissimilarity between each test signature and the set of reference signatures was computed with Eq. 1 or 2. The dissimilarities computed for all the subjects were combined into a ranking list. An equal error rate (EER) was then obtained as the evaluation metric.

3.2. Baselines

We compared our PSN with four baselines, all with the same configuration unless stated otherwise.

DTW. DTW was applied to the 7 of 27 time functions [16] described in Section 2.2.

Table 1. EER (%) comparison for verification of genuine signatures and skilled forgeries. PW and LE denote prewarping and local embedding, respectively. Our method is marked with ⋆. The best performance is highlighted in bold.

Method	PW	LE	MCYT (90%)	MCYT (80%)	MCYT (70%)	FULL
DTW	–	–	4.00	3.00	4.17	2.88
SN	✗	✗	5.50	6.80	6.27	6.72
SN w/ prewarping	✓	✗	6.00	7.00	8.00	6.33
SN w/ location-wise local embedding loss	✗	✓	3.50	3.40	3.75	4.22
⋆ PSN w/ location-wise local embedding loss	✓	✓	**0.50**	2.50	**2.40**	**2.11**
⋆ PSN w/ sequence-wise local embedding loss	✓	✓	**0.50**	**1.75**	2.80	2.90

SN. A Siamese network (SN) was trained with a global contrastive loss (Eq. 7) and has almost the same architecture as PSN in Fig. 2. It has no DTW layer. Its last ReLU is followed by average pooling and l^2-normalization. The final output is a 64D global feature vector.

SN w/ prewarping. The network is basically the same as an SN, but has a DTW layer as in Fig. 2.

SN w/ location-wise local embedding loss. The network is the same as an SN, but was trained with a local embedding loss (Eq. 9).

3.3. Implementation Details

All the compared networks were initialized with He's method [27] and trained for 50 epochs. Each mini-batch consisted of 37 (12 pos & 25 neg) pairs for MCYT-100 and 32 (12 pos & 20 neg) pairs for FULL. Adam [28] was used with an initial learning rate l_0 and an exponential decay $l_0 e^{-0.1i}$ over each epoch i. The momentum and the weight decay were set at 0.9 and 0.0005, respectively. The initial learning rate l_0 and the margin τ were tuned for each method and for each dataset.

During testing, DTW used Eq. 2 as the verifier to handle inter-personal variability in signature length. For all the neural networks, Eq. 1 was used for MCYT-100, while Eq. 2 was used for FULL to handle inter-device variability. The EERs of all the compared methods were obtained by using only the testing set even for DTW.

3.4. Results for Online Signature Verification

Table 1 summarizes the performance of our PSN and the baselines. In general, our method outperformed all the baselines for all the datasets, except for FULL, where the PSN with the sequence-wise loss had a marginally lower EER than DTW. This may be because both the training and the testing data in FULL have very complicated distributions as they contain signatures obtained with four different acquisition protocols. Since the proposed method is learning-based, its performance depends heavily on the quality of the training data, and so suffers more from imbalanced data distributions.

SN underperformed DTW for all the datasets, indicating the inadequacy of the global embedding loss for learning representations of online signatures. As seen from the results of SN w/ prewarping, adding a DTW layer did not improve the SN performance because prewarping has no role in relation to avoiding the loss of temporal information during global pooling. In comparison, replacing the global embedding loss from SN with the location-wise loss yet clearly reduced the EER. Incorporating both prewarping and local embedding into the SN further improved the accuracy because it enabled the SN to be trained with more accurate local correspondences.

It is difficult to reach a categorical decision based on Table 1 as to whether the location-wise loss or the sequence-wise loss is better.

Table 2. EERs (%) published on MCYT. #sub and #ref are the number of subjects and the number of reference signatures, respectively. Our EERs were obtained with location-wise local embedding loss.

Method	#sub	#ref	EER
Fiérrez-Aguilar et al. [13]	145	10	3.36
Argones-Rúa & Alba-Castro [15]	100	10	2.85
Tang et al. [29]	100	10	2.25
Yanikoglu & Kholmatov [7]	100	5	7.80
Vivaracho-Pascual et al. [30]	280	5	6.60
Faúndez-Zanuy [5]	280	5	5.42
Nanni & Lumini [31]	100	5	5.20
Cpalka et al. [32]	100	5	4.88
Sae-Bae & Memon [33]	100	5	4.02
Tang et al. [29]	100	5	3.16
	30	5	2.40
⋆ PSN	20	5	2.50
	10	5	0.50

The location-wise loss performed relatively more consistently. The sequence-wise loss seemed to make our PSN suffer more from inter-device variability. We shall compare them more carefully with more experimental data in the future.

Table 2 summarizes the PSN results together with state-of-the-art EERs previously reported for MCYT. It should be noted that our method is learning-based and so our superior performance was obtained using only the testing set, which contains many fewer subjects. In consequence, the comparison is biased toward our method relative to the state of the art. In the future, we shall consider training the network with a larger, potentially synthetic, dataset and testing it on the whole of each public dataset.

4. CONCLUSION

We proposed incorporating an SN into DTW to learn local representations for online signature verification. Compared with a conventional SN, the proposed PSN is invariant when faced with temporal distortions and simultaneously imposes a temporal consistency constraint on verification. The experimental results showed that the temporal invariance was not compatible with global embedding but clearly reduced the error rate in combination with a local embedding loss. Meanwhile, the temporal consistency constraint consistently improved the accuracy whether temporal invariance is considered or not. In the future, we shall make use of larger datasets in combination with the PSN proposed in this paper.

5. REFERENCES

[1] K. Sundararajan and D. L. Woodard, "Deep learning for biometrics: A survey," *ACM Comput. Surv.*, vol. 51, no. 3, pp. 65:1–65:34, 2018.

[2] M. I. Malik, S. Ahmed, A. Marcelli, et al., "ICDAR2015 competition on signature verification and writer identification for on- and off-line skilled forgeries (SigWIcomp2015)," in *ICDAR*, 2015, pp. 1186–1190.

[3] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 26, no. 1, pp. 43–49, 1978.

[4] A. Kholmatov and B. A. Yanikoglu, "Identity authentication using improved online signature verification method," *Pattern Recognition Letters*, vol. 26, no. 15, pp. 2400–2408, 2005.

[5] M. Faúndez-Zanuy, "On-line signature recognition based on VQ-DTW," *Pattern Recognition*, vol. 40, no. 3, pp. 981–992, 2007.

[6] A. Kholmatov and B. A. Yanikoglu, "SUSIG: An on-line signature database, associated protocols and benchmark results," *Pattern Anal. Appl.*, vol. 12, no. 3, pp. 227–236, 2009.

[7] B. A. Yanikoglu and A. Kholmatov, "Online signature verification using Fourier descriptors," *EURASIP J. Adv. Sig. Proc.*, 2009.

[8] M. D. Cabrera, A. Fischer, R. Plamondon, and M. A. Ferrer, "Towards an automatic on-line signature verifier using only one reference per signer," in *ICDAR*, 2015, pp. 631–635.

[9] A. Fischer, M. D. Cabrera, R. Plamondon, and M. A. Ferrer, "Robust score normalization for DTW-based on-line signature verification," in *ICDAR*, 2015, pp. 241–245.

[10] M. Gomez-Barrero, J. Galbally, J. Fiérrez, J. Ortega-Garcia, and R. Plamondon, "Enhanced on-line signature verification based on skilled forgery detection using Sigma-LogNormal features," in *ICB*, 2015, pp. 501–506.

[11] R. Tolosana, R. Vera-Rodríguez, J. Ortega-Garcia, and J. Fiérrez-Aguilar, "Preprocessing and feature selection for improved sensor interoperability in online biometric signature verification," *IEEE Access*, vol. 3, pp. 478–489, 2015.

[12] M. D. Cabrera, A. Fischer, M. A. Ferrer, and R. Plamondon, "Dynamic signature verification system based on one real signature," *IEEE Trans. Cybernetics*, vol. 48, no. 1, pp. 228–239, 2018.

[13] J. Fiérrez-Aguilar, J. Ortega-Garcia, D. Ramos, and J. Gonzalez-Rodriguez, "HMM-based on-line signature verification: Feature extraction and signature modeling," *Pattern Recognition Letters*, vol. 28, no. 16, pp. 2325–2334, 2007.

[14] J. Galbally, J. Fiérrez, M. Martinez-Diaz, and J. Ortega-Garcia, "Improving the enrollment in dynamic signature verification with synthetic samples," in *ICDAR*, 2009, pp. 1295–1299.

[15] E. Argones-Rúa and J. L. Alba-Castro, "Online signature verification based on generative models," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 42, no. 4, pp. 1231–1242, 2012.

[16] M. Martinez-Diaz, J. Fiérrez, R. P. Krish, and J. Galbally, "Mobile signature verification: Feature robustness and performance comparison," *IET Biometrics*, vol. 3, no. 4, pp. 267–277, 2014.

[17] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a Siamese time delay neural network," in *NIPS*, 1993, pp. 737–744.

[18] S. Lai, L. Jin, and W. Yang, "Online signature verification using recurrent neural network and length-normalized path signature descriptor," in *ICDAR*, 2017, pp. 400–405.

[19] K. Ahrabian and B. Babaali, "Usage of autoencoders and Siamese networks for online handwritten signature verification," *Neural Computing and Applications*, 2018.

[20] W. Pei, *Models for supervised learning in sequence data*, Ph.D. thesis, Delft University of Technology, Netherlands, 2018.

[21] R. Tolosana, R. Vera-Rodríguez, J. Fiérrez, and J. Ortega-Garcia, "Exploring recurrent neural networks for on-line handwritten signature biometrics," *IEEE Access*, vol. 6, pp. 5128–5138, 2018.

[22] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015, pp. 448–456.

[23] F. Radenovic, G. Tolias, and O. Chum, "Fine-tuning cnn image retrieval with no human annotation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[24] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *CVPR*, 2006, pp. 1735–1742.

[25] J. Ortega-Garcia, J. Fierrez-Aguilar, D. Simon, et al., "MCYT baseline corpus: A bimodal biometric database," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 150, no. 6, pp. 395–401, 2003.

[26] J. Galbally, M. D. Cabrera, M. A. Ferrer-Ballester, et al., "On-line signature recognition through the combination of real dynamic data and synthetically generated static data," *Pattern Recognition*, vol. 48, no. 9, pp. 2921–2934, 2015.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *ICCV*, 2015, pp. 1026–1034.

[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[29] L. Tang, W. Kang, and Y. Fang, "Information divergence-based matching strategy for online signature verification," *IEEE Trans. Information Forensics and Security*, vol. 13, no. 4, pp. 861–873, 2018.

[30] C. Vivaracho-Pascual, M. Faúndez-Zanuy, and J. M. Pascual, "An efficient low cost approach for on-line signature recognition based on length normalization and fractional distances," *Pattern Recognition*, vol. 42, no. 1, pp. 183–193, 2009.

[31] L. Nanni and A. Lumini, "A novel local on-line signature verification system," *Pattern Recognition Letters*, vol. 29, no. 5, pp. 559–568, 2008.

[32] K. Cpalka, M. Zalasinski, and L. Rutkowski, "A new algorithm for identity verification based on the analysis of a handwritten dynamic signature," *Appl. Soft Comput.*, vol. 43, pp. 47–56, 2016.

[33] N. Sae-Bae and N. D. Memon, "Online signature verification on mobile devices," *IEEE Trans. Information Forensics and Security*, vol. 9, no. 6, pp. 933–947, 2014.