Deep Attentive Time Warping

Shinnosuke Matsuo[a,*], Xiaomeng Wu[b], Gantugs Atarsaikhan[c], Akisato Kimura[b], Kunio Kashino[b], Brian Kenji Iwana[a], Seiichi Uchida[a]

[a] *Department of Advanced Information Technology, Kyushu University, Fukuoka, Japan*
[b] *Communication Science Laboratories, NTT Corporation, Kanagawa, Japan*
[c] *Institute for Molecular Medicine Finland, HiLIFE, University of Helsinki, Helsinki, Finland*

Abstract

Similarity measures for time series are important problems for time series classification. To handle the nonlinear time distortions, Dynamic Time Warping (DTW) has been widely used. However, DTW is not learnable and suffers from a trade-off between robustness against time distortion and discriminative power. In this paper, we propose a neural network model for task-adaptive time warping. Specifically, we use the attention model, called the bipartite attention model, to develop an explicit time warping mechanism with greater distortion invariance. Unlike other learnable models using DTW for warping, our model predicts all local correspondences between two time series and is trained based on metric learning, which enables it to learn the optimal data-dependent warping for the target task. We also propose to induce pre-training of our model by DTW to improve the discriminative power. Extensive experiments demonstrate the superior effectiveness of our model over DTW and its state-of-the-art performance in online signature verification.

Keywords: Dynamic time warping, attention model, metric learning, time series classification, online signature verification

[*] Corresponding author

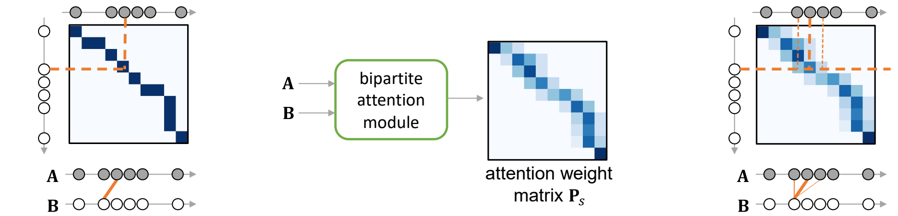
Email addresses: shinnosuke.matsuo@human.ait.kyushu-u.ac.jp (Shinnosuke Matsuo), uchida@ait.kyushu-u.ac.jp (Seiichi Uchida)

1. Introduction

Measuring similarity is one of the most important tasks for time series recognition. For example, similarity gives an essential criterion for classifying time series. Many applications, such as activity recognition, computational auditory scene analysis, computer security, electronic health records, and biometrics (e.g., handwritten signature verification) [1], use time series similarity for recognition. One difficulty in measuring similarity is due to nonlinear time distortions. The distortions can appear as temporal shifts, stretches and contractions, and other various nonlinear temporal fluctuations.

To be invariant to nonlinear time distortions, Dynamic Time Warping (DTW) [2] has been widely utilized. Let $\mathbf{A} = \mathbf{a}_1, \ldots, \mathbf{a}_i, \ldots, \mathbf{a}_I$ and $\mathbf{B} = \mathbf{b}_1, \ldots, \mathbf{b}_j, \ldots, \mathbf{b}_J$ denote two time series, where both $\mathbf{a}_i$ and $\mathbf{b}_j$ are D-dimensional feature vectors. As shown in Fig. 1(a), DTW establishes a "hard" correspondence between $\mathbf{A}$ and $\mathbf{B}$ as a path on a two-dimensional plane, or an $I \times J$ binary matrix. Here the term "hard" implies "one or zero;" the (i, j)-th element of the matrix becomes 1, if $\mathbf{a}_i$ and $\mathbf{b}_j$ is "matched (i.e., corresponding)," and zero, otherwise. The correspondence is determined to minimize the distance $\mathbf{A}$ and $\mathbf{B}$ by dynamic programming (DP).

In DTW, several hand-crafted constraints are often assumed for controlling the correspondence. Traditionally, monotonicity, continuity, and boundary con-


A
B
bipartite attention module
A
B
attention weight matrix $\mathbf{P}_s$
A
B

(a) Hard correspondence by DTW path

(b) Proposed (Deep attentive time warping)

(c) Soft correspondence by attention weight matrix

Figure 1: (a) DTW conducts a hard correspondence between two sequences $\mathbf{A}$ and $\mathbf{B}$. (b) The proposed *deep attentive time warping* is composed of a bipartite attention module, which generates an attention weight matrix $\mathbf{P}_s$. (c) The attention weight matrix $\mathbf{P}_s$ represents soft correspondence between two sequences $\mathbf{A}$ and $\mathbf{B}$.

straints have been utilized. These constraints are appropriate and acceptable in many applications but encounter trade-off problems in the warping flexibility. If the constraints are too loose, DTW causes "over-warping" that cancels important inter-class differences and loses its discriminative power. If we add more constraints heuristically (like [3, 4]) to avoid over-warping, DTW then can not remove intra-class distortions sufficiently.

In recent years, *deep metric learning* has been applied to various classification tasks. Deep metric learning is a machine learning technique to learn an adaptive feature space that takes into account the similarity (or dissimilarity) relationships among data [5–8]. In the typical formulation, a Siamese or triplet neural network is trained to learn an embedding space, where closeness between embeddings (i.e., features extracted from the network) encodes the level of similarities between the data samples. It enforces the embeddings to lie close if the samples belong to the same class, and pushes them apart if different.

Deep metric learning for image classification tasks has had many successful results [9–11]. Whereas for time series, there is still room for improvement. As detailed in Section 2, the past attempts either suffer from the loss of useful temporal information [12–14] or are not explicitly invariant to nonlinear time distortions [15–17].

In this paper, we propose a novel neural network-based time warping model, called *deep attentive time warping*[1]. The proposed method is based on a novel learnable time warping mechanism with contrastive metric learning. Its key idea is a novel attention model, called the *bipartite attention module*. As shown in Fig. 1 (b), this module takes two time series as inputs and predicts an *attention weight matrix*. This matrix represents the "soft" correspondence between all time steps of the two inputs, as shown in Fig. 1 (c). By training the bipartite attention module appropriately for a specific task, we can realize time warping that can mitigate the trade-off between robustness against time distortion and

[1] The code is available at `https://github.com/matsuo-shinnosuke/deep-attentive-time-warping`.

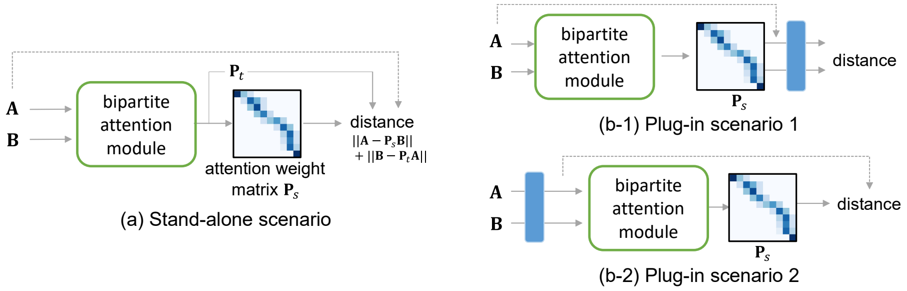

A
B
bipartite attention module
P_t
attention weight matrix P_s
distance
||A − P_sB|| + ||B − P_tA||
(a) Stand-alone scenario
A
B
bipartite attention module
P_s
distance
(b-1) Plug-in scenario 1
A
B
bipartite attention module
P_s
distance
(b-2) Plug-in scenario 2

Figure 2: The proposed deep attentive time warping can be used in two scenarios. One is a stand-alone scenario (a), and the other is a plug-in scenario (b-1) and (b-2). In (b-1) and (b-2), blue boxes represent a neural network for contrastive representation learning.

discriminative power. In other words, the learned soft correspondence will enhance important inter-class differences and, at the same time, remove intra-class distortions.

The proposed method has great versatility and can be used for applications in two different scenarios; one is a stand-alone scenario and the other is a plug-in scenario. As shown in Fig. 2 (a), the former takes two inputs **A** and **B** and determines their difference by utilizing their original feature representation. In the latter scenario, we use existing contrastive metric learning frameworks with the standard DTW. Then, as shown in Figs. 2 (b-1) and (b-2), we replace the DTW with the proposed method. Consequently, our deep attentive time warping is combined with contrastive representation learning and the entire framework becomes totally trainable for better (i.e., contrastive) time warping and feature representation.

We conduct extensive experiments to demonstrate the superior effectiveness of the proposed method. We first conduct two experiments in the stand-alone scenario to confirm how the proposed method provides reasonable time warping for the classification. We prove that the proposed method achieves better classification performance with effective warping than the other time warping techniques through qualitative and quantitative evaluations on the well-known Unipen [18] and the University of California Riverside (UCR) [19] datasets. We then conduct another experiment in the plug-in scenario. Specifically, through an online signature verification experiment, we prove that the proposed method achieves state-of-the-art performance by outperforming other learnable time warping methods.

The main contributions of this paper are summarized as follows:

- A novel neural network-based time warping method, called deep attentive time warping, is proposed by introducing a bipartite attention module. It is learnable, task-adaptive, and improves the trade-off between robustness against time distortion and discriminative power. We also prove a two-step training process enhances the performance.

- We show the high versatility of the proposed deep attentive time warping by using it in two different scenarios, stand-alone and plug-in.

- Extensive experiments on more than 50 public datasets demonstrate the superior effectiveness of the proposed method over DTW as a stand-alone time warping model.

- We experimentally show that the proposed method in the plug-in scenario achieves better performance than state-of-the-art learnable time warping methods in an online signature verification task.

2. Related Work

2.1. Dynamic time warping

DTW [2] (standard DTW) is a time warping method that has been used for a long time as a time series similarity measure invariant to nonlinear time distortion. As noted in Section 1, DTW can determine the hard correspondence between **A** and **B**. While DTW exhibits great distortion invariance, it may cause over-warping that often results in incorrect classification.

There are many attempts to improve the performance of DTW. To suppress over-warping, early studies [2, 3] proposed to put a warping window as an additional constraint to the standard monotonicity and continuity constraints. Roughly speaking, $\mathbf{a}_i$ is able to match with one of $\mathbf{b}_{i-w}, \ldots, \mathbf{b}_i, \ldots, \mathbf{b}_{i+w}$, where w is the window width. A smaller w will have fewer over-warping cases. In [20, 21], the warping path is penalized by the difference of i and j of the matched $\mathbf{a}_i$ and $\mathbf{b}_j$. Soft-DTW [22] is an interesting attempt to replace the min operation with a soft-min operation, which is realized by logarithmic and exponential functions. With this replacement, DTW becomes differentiable and can be built in various machine learning frameworks.

2.2. DTW with deep metric learning

In recent years, more efforts have been made on deep metric learning for time series [12–17]. They are based on a feature extraction mechanism with a Siamese network, which is trained by a loss function evaluating the distance between the features. The extracted features from time series are either global or local. Compared to the standard DTW, these methods achieve better accuracy; however, they do not treat the temporal distortion explicitly. This means that they do not warp one time series to another and, thus, is impossible to introduce an explicit control of warping flexibility.

Several metric learning methods introduce DTW for an explicit removal of temporal distortions. More specifically, they introduce the standard DTW before or after a Siamese network. Prewarping Siamese Network (PSN) [23] and Time Aligned Recurrent Neural Networks (TARNN) [24] perform DTW between two time series and then fed the warped result to a Siamese network for metric learning. In contrast, Deep DTW (DDTW) [25] first extracts a sequence of local features from each time series and then performs DTW. With the introduction of DTW, these methods could achieve better performance than simple metric learning methods. Note that they do not learn the warping characteristics; their temporal distortion removal ability relies on the standard DTW.

Few methods [26, 27] have been proposed for learning warping characteristics. They calculate a quasi-binary *matchability* $\Phi(i, j)$ between each (i, j) pair. Then, all IJ point-wise distances between $i \in [1, I]$ and $j \in [1, J]$ are aggregated by using $\Phi(i,j)$; if $\Phi(i,j) \sim 1$, the distance between i and j is taken into account. Since the matchability is determined *independently* for each (i,j) pair by a neural network, it is time-consuming to have all IJ matchability results. More importantly, this independent determination process cannot control the global warping characteristics, which have been carefully treated even in the standard DTW.

Furthermore, there is a preliminary conference paper of this work [28] and this paper contains significant differences from it. First, we newly propose a plug-in scenario, where our deep attentive time warping is utilized as a differentiable module in a large classification system. We further confirmed that the plug-in usage of our technique achieves the state-of-the-art performance in large-scale signature verification tasks. Moreover, for the stand-alone scenario, we conduct more extensive comparison experiments on over 50 classification tasks in UCR dataset, whereas only four tasks have been tackled in [28]. Technical details are also newly elaborated in this paper.

3. Deep Attentive Time Warping

3.1. Overview

We propose *deep attentive time warping*, a novel neural network-based time warping method. As noted in Section 1, the proposed method can be used to evaluate the distance/dissimilarity between two time series (e.g., series of raw signals or deep features) $\mathbf{A}$ and $\mathbf{B}$ in its stand-alone scenario of Fig. 2 (a). It also can be used as an attention-based feature extractor in its plug-in scenario, as shown in Fig. 2 (b).

As shown in Figs. 1 (b) and (c), the *bipartite attention module* generates the attention weight matrix $\mathbf{P}_s$, which represents time warping between $\mathbf{A}$ and $\mathbf{B}$ as a soft temporal correspondence. The bipartite attention module is trained by metric learning with contrastive loss. The resulting matrix $\mathbf{P}_s$ is expected to provide not only distortion invariance but also discriminative power, both of which are appropriate for the target task.

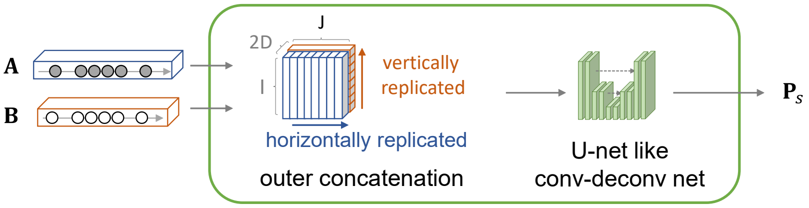
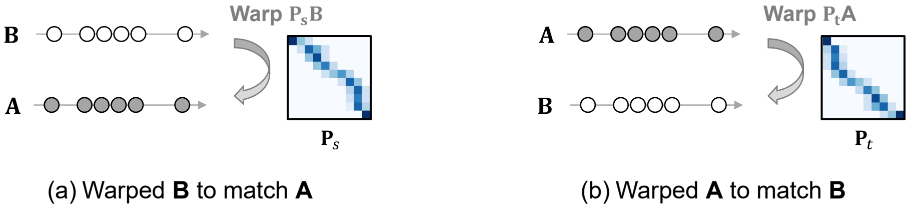


Figure 3: Overview of the bipartite attention module.



(a) Warped **B** to match **A**

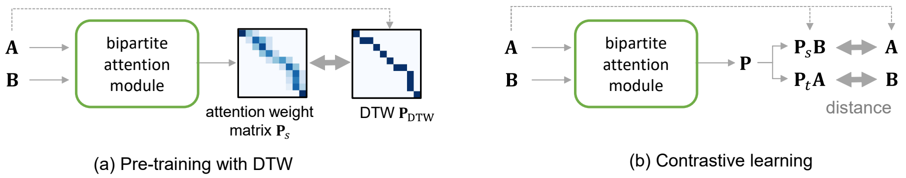
(b) Warped **A** to match **B**

Figure 4: The time warping by the attention weight matrix.

3.2. Time warping with the bipartite attention module

The detail of the bipartite attention module is shown in Fig. 3. In the bipartite attention module, two multivariate time series $\mathbf{A}$ and $\mathbf{B}$ are first combined by "outer concatenation" to have a two-dimensional array of the concatenated vectors of $\mathbf{a}_i$ and $\mathbf{b}_j$ (i.e., a third-order tensor). Specifically, by replicating $\mathbf{A}$ horizontally J times and $\mathbf{B}$ vertically I times, we have two $I \times J \times D$ tensors and concatenate them as an $I \times J \times 2D$ tensor. The tensor is then input to a Fully Convolutional Network (FCN) which functions as an attention model. In this paper, we utilize U-Net as an FCN.

Before outputting an attention weight matrix $\mathbf{P}_s$, a row-wise softmax operation is applied to the output of the FCN, so that the sum of the values in the rows becomes 1. This operation is important for using $\mathbf{P}_s$ as the soft-correspondence, as shown in Fig. 1 (c). Consequently, the attention weight matrix $\mathbf{P}_s$ is used for warping of $\mathbf{B}$, as shown in Fig. 4 (a). The time warping of $\mathbf{B}$ is simply given by the matrix product $\mathbf{P}_s\mathbf{B}$ and expected to be similar to $\mathbf{A}$. In a similar way, we also have another matrix $\mathbf{P}_t$, which warps $\mathbf{A}$ to be $\mathbf{P}_t\mathbf{A} \sim \mathbf{B}$, as shown in


A
B
bipartite attention module
attention weight matrix $\mathbf{P}_s$
DTW $\mathbf{P}_{\mathrm{DTW}}$
A
B
bipartite attention module
P
$\mathbf{P}_s\mathbf{B}$
A
$\mathbf{P}_t\mathbf{A}$
B
distance

(a) Pre-training with DTW

(b) Contrastive learning

Figure 5: The bipartite attention module is optimized in a two-step manner. In the first step, The module is pre-trained to mimic DTW, and in the second step, the module is optimized by contrastive training.

Fig. 4 (b). The matrix $\mathbf{P}_t$ is given by first transposing the output of FCN and then applying the row-wise softmax operation.

As clarified above, the matrix $\mathbf{P}_s$ (and $\mathbf{P}_t$) is used as an attention for controlling the time series $\mathbf{B}$ ($\mathbf{A}$) to be similar to $\mathbf{A}$ ($\mathbf{B}$). The bipartite attention module drives the matrices $\mathbf{P}_s$ and $\mathbf{P}_t$ at the same time by utilizing two-dimensional nature of the outer-concatenated representation of $\mathbf{A}$ and $\mathbf{B}$. In other words, $\mathbf{A}$ is used to attend individual elements of $\mathbf{B}$ and vice versa. This mutual attention is analogous to the cost matrix of the so-called bipartite matching problem. We, therefore, call our special attention scheme bipartite attention and differentiate from popular attention schemes such as additive attention [29] and dot-product attention [30, 31].

Since we use an FCN (U-Net) in the bipartite attention module, the proposed method, theoretically, can handle time series samples with variable lengths. Namely, the lengths I and J can be different among samples. In the later experiments, however, we use a fixed-length time series by following the traditional experimental setup of the comparative methods (such as DDTW and PSN). This fixed-length condition also allows efficient batch-based training.

3.3. Learning attention model with contrastive loss

To achieve time warping with sufficient time distortion invariance and discriminative power for a specific task, we learn the bipartite attention model with the following *dual contrastive loss*:

$$\mathcal{L}(\mathbf{A},\mathbf{B}) = \mathcal{L}_s(\mathbf{A},\mathbf{B}) + \mathcal{L}_t(\mathbf{A},\mathbf{B}). \tag{1}$$

Both $\mathcal{L}_s$ and $\mathcal{L}_t$ are formulated as a contrastive loss [32] specialized for the proposed method. More specifically, $\mathcal{L}_s$ is formulated as

$$\mathcal{L}_s(\mathbf{A}, \mathbf{B}) = \begin{cases} \frac{1}{ID}\|\mathbf{A} - \mathbf{P}_s\mathbf{B}\|_{\mathrm{F}}^2 & \text{if a same-class pair,} \\ \max\left(0, \tau - \frac{1}{ID}\|\mathbf{A} - \mathbf{P}_s\mathbf{B}\|_{\mathrm{F}}^2\right) & \text{otherwise,} \end{cases} \tag{2}$$

where τ is the hyper-parameter for margin. $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius norm. If $\mathbf{A}$ and $\mathbf{B}$ are a same-class pair, the distance between the input $\mathbf{A}$ and the warped input $\mathbf{P}_s\mathbf{B}$ is minimized. If not, their distance is optimized to be larger than τ. The other loss $\mathcal{L}_t$ is defined by using $\|\mathbf{B} - \mathbf{P}_t\mathbf{A}\|_{\mathrm{F}}$ and J (the length of $\mathbf{B}$), instead. Fig. 5 (b) summarizes the above process to train the bipartite attention module.

It should be emphasized that the above contrastive learning has a clear advantage over the standard DTW. The objective of the standard DTW is to minimize the distance between two time series regardless of whether they belong to the same class or not. Therefore, the standard DTW often underestimates the distance for different-class pairs. In contrast, the proposed method considers their classes and therefore can have appropriate time distortion invariance and discriminability at the same time.

3.4. Pre-training with the standard DTW

The proposed deep attentive time warping has much more warping flexibility than the standard DTW. As reviewed in 2.1, several constraints, such as monotonicity and continuity, are imposed to control the warping path in the standard DTW. Since the proposed method does not have such constraints, it has higher flexibility. However, of course, too much flexibility is not appropriate for many applications.

We, therefore, introduce a pre-training phase with the standard DTW so that the proposed method can mimic the DTW before starting its main training phase. Specifically, as shown in Fig. 5 (a), we prepare a binary matrix $\mathbf{P}_{\mathrm{DTW}}$ showing the DTW path between $\mathbf{A}$ and $\mathbf{B}$, then pre-train the FCN to minimize the following loss function:

$$\mathcal{L}_{\mathrm{pre}}(\mathbf{P}_s, \mathbf{P}_{\mathrm{DTW}}) = \frac{1}{IJ} \|\mathbf{P}_s - \mathbf{P}_{\mathrm{DTW}}\|_{\mathrm{F}}^2. \tag{3}$$

After the above pre-training phase, the attention model is further trained using the contrastive loss, as described in Section 3.3. By this two-step training scheme, the proposed method can avoid excessive warping flexibility, while keeping more flexibility than the standard DTW. We confirm the positive effect of pre-training through ablation studies in later experiments.

4. Preliminary Experiments in Stand-Alone Scenario

We conducted the experimental evaluation of the proposed deep attentive time warping in the stand-alone scenario. As shown in Fig. 2 (a), this scenario uses the proposed method for calculating a distance between two time series and then the distance can be used in, for example, a nearest-neighbor classifier. First, we conduct qualitative evaluations and show the behavior of the proposed method on the online handwritten character dataset, called Unipen, as a simple example. Next, we conduct quantitative evaluations and show the proposed method has a better trade-off between robustness against time distortion and discriminative power than DTW on 52 datasets of the famous UCR Archive.

Note that the experiments in this section mainly aim to confirm the time warping ability of the proposed method, and thus comparative study will be made with rather traditional DTW methods. The comparisons with state-of-the-art learnable time warping methods will be shown in the next section.

4.1. Qualitative evaluations using online handwritten samples

4.1.1. Unipen Dataset

Unipen [18] is comprised of several subsets and we used the most popular ones, Unipen 1a (digits, 10 classes, 7,562 samples in total), Unipen 1b (uppercase alphabet, 26 classes, 6,039 samples), and Unipen 1c (lowercase alphabet, 26 classes, 10,712 samples), for the evaluation. Each sample is a sequence of

2D pen-tip coordinate vectors. For the detailed comparison with fixed-length methods (such as SVM, 1D-CNN, and Siamese), linear resampling is performed on each sample so that their temporal length was 50. The 2D coordinates were normalized to the range $[-1, 1]$. For each class, 200 samples were randomly selected for validation, and other 200 for test. All the remaining samples were used for training.

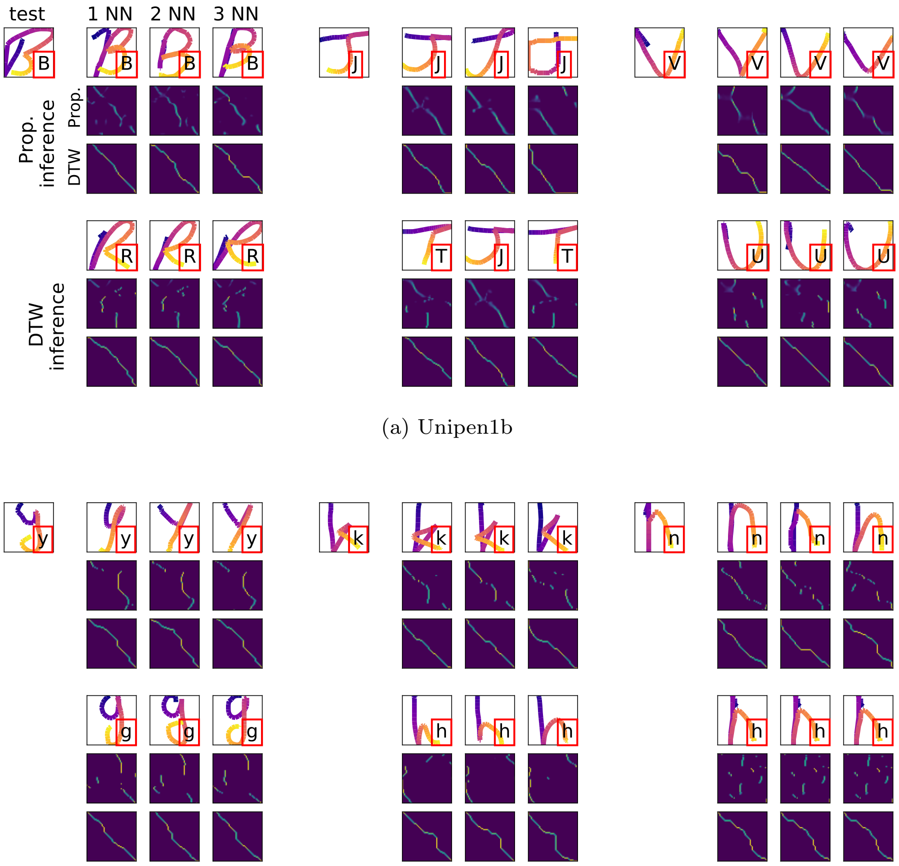
4.1.2. Implementation details

The network architecture in the bipartite attention module follows the original U-net [33], except for an additional batch normalization layer after each convolutional layer. The learning rate was set to 0.0001, and Adam [34] was used as the optimizer. Before pre-training, the network weights were initialized by He initialization [35]. The batch size was set to 512. During training, same-class and different-class pairs were loaded in a ratio of $1 : 2$. The hyperparameter τ in the contrastive loss was set to 1. The maximum iterations for pre-training of Fig. 5 (a) and the main contrastive training of (b) were set at $1,000$ and $10,000$, respectively; and the best model (i.e., the best iteration number) was chosen by the evaluation with the validation set.

For quantitative evaluation, we conducted a classification experiment using the distance by the proposed method. For each test sample, its distances to all training samples were calculated by the proposed method and the k-nearest neighbor classification was performed to determine its class label at$k = 1$. As the distance between $\mathbf{A}$ and $\mathbf{B}$, we use the following "symmetric" distance:

$$d(\mathbf{A}, \mathbf{B}) = \frac{1}{ID}\|\mathbf{A} - \mathbf{P}_s\mathbf{B}\|_\mathrm{F}^2 + \frac{1}{JD}\|\mathbf{B} - \mathbf{P}_t\mathbf{A}\|_\mathrm{F}^2. \tag{4}$$

The proposed method achieved 99.0, 98.0, and 95.5% classification accuracies for Unipen 1a, 1b, and 1c, respectively, whereas the standard DTW achieved 98.4, 96.0, and 94.1%. This proves the proposed method achieved sufficient accuracies and, therefore, the following qualitative evaluation results are reliable enough.



(a) Unipen1b

(b) Unipen1c

Figure 6: A visualization of the matching path of the improved samples compared with DTW. The character in the red box shows the ground truth.

4.1.3. Qualitative evaluation results

Figs. 6 (a) and (b) show the results on three test samples of Unipen 1b and 1c, respectively. Those test samples are correctly classified by the proposed method and not by DTW. For each test sample, the top three nearest neighbors by distance of the proposed method and those by the DTW distance are shown. The attention weight matrices and the DTW matching paths are also shown.

In Fig. 6 (a), the proposed method classifies the test sample as ‘B’ correctly by attention matrices that resemble the DTW path. In contrast, DTW clas-

Table 1: Error rates (%) between confusing classes (Unipen). Error rates in red indicate the least rate of each case.

	Unipen 1b		Unipen 1c		
Method	‘J’ vs. ‘T’	‘U’ vs. ‘V’	‘g’ vs. ‘y’	‘h’ vs. ‘k’	‘h’ vs. ‘n’
ours	3.5	6.5	3.0	3.0	6.5
DTW	14.0	11.5	12.0	8.0	10.0

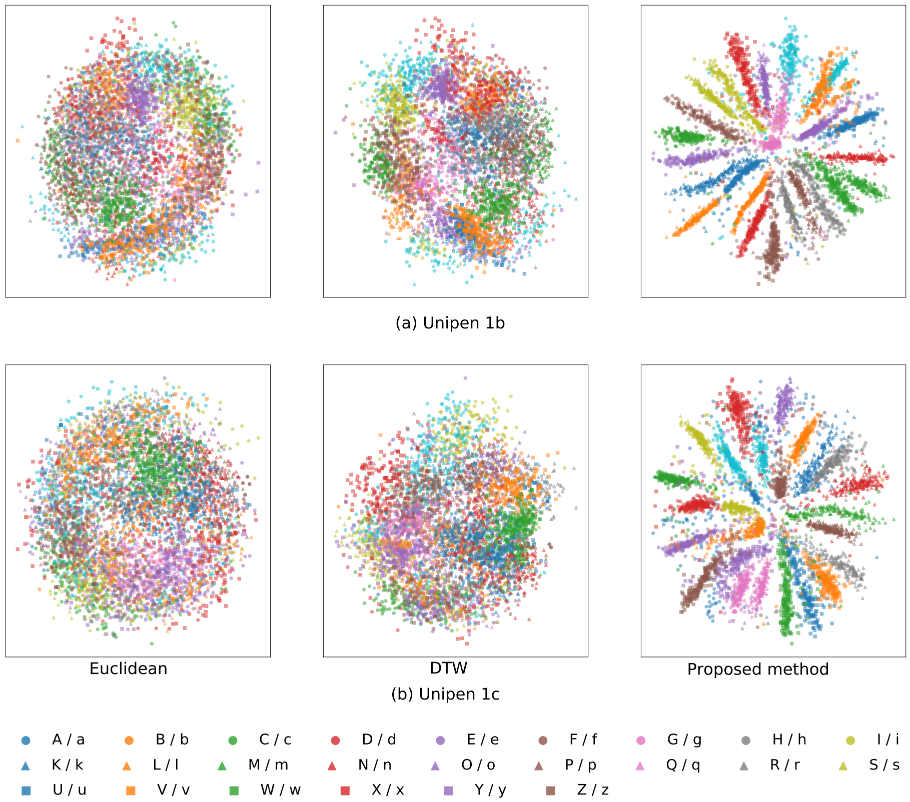
sifies this ‘B’ as ‘R’ incorrectly. It should be emphasized that the proposed method provided an almost meaningless attention matrix between ‘B’ and ‘R,’ *intentionally.* This is because the proposed method tries to differentiate them as an expected effect of its contrastive learning. Similar attention matrices are found in other cases. Since DTW has no such function, it always gives smooth correspondence and gets a small distance that causes misclassification.

The third nearest neighbor of ‘J’ in the middle column of Fig. 6 (a) shows another benefit of the proposed method. This ‘J’ shows a different stroke order. Since the proposed method does not have a strict monotonicity constraint, its attention map deals with the stroke order variation. From these results, we can observe that the proposed method has an appropriate time warping flexibility that realizes both sufficient time distortion invariance and discriminability.

Fig. 7 shows the distribution of test samples by the multi-dimensional scaling (MDS). Three distance metrics, Euclidean, DTW, and the proposed method, are used for these MDS visualizations. For the proposed method, the distance between a pair of test samples $\mathbf{A}$ and $\mathbf{B}$ is evaluated by (4).

These distributions prove that the distance by DTW is more discriminative than Euclidean, and the distance by the proposed method is far more discriminative than DTW. For example, the overlap between ‘U’ and ‘V’ in Unipen 1b by the DTW distance disappears in the proposed method. The contrastive metric learning in the proposed method realizes this discriminability, as expected.

Table 1 shows the error rates for binary classifications between ambiguous class pairs in Unipen 1b and 1c. Fig. 8 shows the normalized histograms of the



(a) Unipen 1b

Euclidean

DTW

Proposed method

(b) Unipen 1c

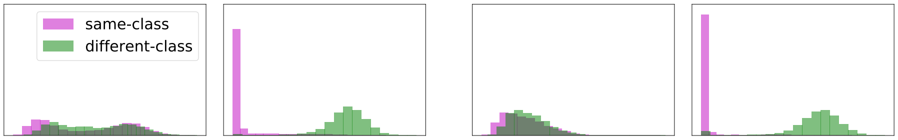
Figure 7: The visualization of the test samples by MDS on (a) Unipen 1b and (b) 1c.

distances by DTW and the proposed method for the ambiguous class pairs in Unipen 1b. The histogram of DTW shows a large overlap between the same and different-class pairs, whereas the proposed method does not. These results also prove the sufficient discriminative power of the proposed method.

4.2. Quantitative analysis on UCR dataset

4.2.1. UCR Dataset

The University of California Riverside (UCR) Time Series Classification Archive (2015 edition) [19] is a famous benchmark that is comprised of 85 different univariate time series datasets. Among them, we selected 52 datasets satisfying the following two conditions. The first condition requires that more than 100 training samples are available. The second requires that the sample



(a) ‘J’ vs. ‘T’

(b) ‘U’ vs. ‘V’

Figure 8: Distance histograms for two confusing class pairs, ‘J’ vs. ‘T’ and ‘U’ vs. ‘V,’ of Unipen 1b. For each pair, the left histogram is about DTW distance and the right is the proposed method.

length (I and J) should be less than 1,000. The proposed method can, theoretically, deal with any sample length (even longer than 1,000); however, in practice, too long of samples cause memory issues (like other trainable time warping methods). In each dataset, all the samples are already regulated to have the same length. UCR prepares a training sample set and a test sample set for each dataset. Among the training samples, 90% is used for training and 10% for validation. All time series were standardized for each channel to have a mean of zero and a variance of one.

4.2.2. Implementation details

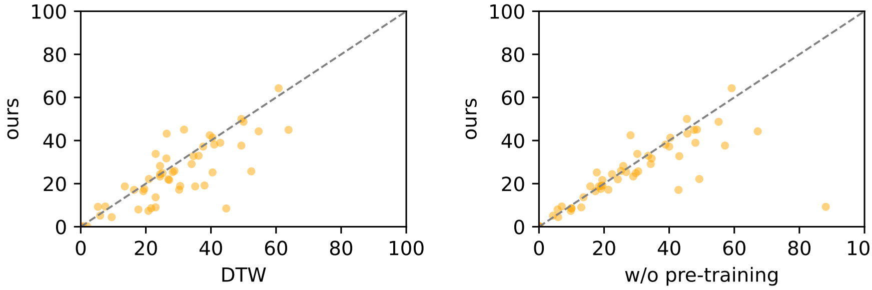
The model architecture, learning rate, optimizer, hyper-parameter, number of iterations, and inference protocol are the same as 4.1.2. The batch size was determined for the maximum memory utilization of the GPU (Tesla V100).

We compared the proposed method with the standard DTW (DTW) [2], window-DTW (w-DTW) [2] and soft-DTW (s-DTW) [22]. The optimal values of the hyperparameters in the comparative methods, as well as the proposed method, were chosen by the validation set[2]. As noted above, 10% of UCR training set were used as the validation set. For example, the hyperparameter γ in s-DTW was chosen from 0.01, 0.1, 1, 10, 100 using the validation set.

[2] Exceptionally, the value of the hyperprameter “window size” in w-DTW was taken from the list of https://www.cs.ucr.edu/~eamonn/time_series_data_2018/.

Table 2: Error rates by the proposed method (ours) and the comparative methods. Error rates in red and blue indicate the least and the second least rates, respectively. If an error rate of a comparative method is printed in **bold**, the proposed method was superior to the comparative method with statistical significance at the 5% level by McNemar's test. If *italic*, the proposed method is inferior with significance at the 5% level.

Dataset	Train	Test	Class	Length	DTW	w-DTW	s-DTW	ours	w/o pre-train.
Adiac	390	391	37	176	39.64	38.87	41.94	42.46	*28.13*
ChlorineConcentration	467	3840	3	166	**35.16**	**35.16**	**35.42**	18.72	18.41
Computers	250	250	2	720	37.60	43.60	34.80	37.20	40.00
CricketX	390	390	12	300	24.36	24.87	*22.82*	28.21	25.90
CricketY	390	390	12	300	25.13	25.13	25.90	24.87	29.74
CricketZ	390	390	12	300	24.36	**29.49**	21.79	23.33	**28.97**
DistalPhalanxOutlineAgeGroup	400	139	3	80	*23.02*	*23.02*	23.74	33.81	30.22
DistalPhalanxOutlineCorrect	600	276	2	80	28.26	27.54	23.91	25.36	26.81
DistalPhalanxTW	400	139	6	80	41.01	41.01	39.57	38.13	38.85
Earthquakes	322	139	2	512	28.78	31.65	22.30	25.90	25.18
ECG200	100	100	2	96	**23.00**	**23.00**	18.00	9.00	13.00
ECG5000	500	4500	5	140	*7.56*	*7.49*	*7.27*	9.40	*7.00*
ElectricDevices	8926	7711	7	96	40.50	*38.28*	*39.37*	41.33	*40.37*
FaceAll	560	1690	14	131	**19.23**	**18.28**	**20.24**	16.51	17.34
FacesUCR	200	2050	14	131	**9.51**	**8.39**	**8.63**	4.44	**5.90**
FiftyWords	450	455	50	270	**30.55**	**27.69**	20.44	18.90	19.12
Fish	175	175	7	463	**17.71**	**16.57**	13.71	8.00	5.71
FordA	3601	1320	2	500	**44.70**	**32.20**	**39.02**	8.48	10.00
FordB	3636	810	2	500	**38.02**	**37.90**	**41.98**	19.14	19.51
Ham	109	105	2	431	**52.38**	**52.38**	**41.90**	25.71	30.48
InsectWingbeatSound	220	1980	11	256	**63.84**	42.98	44.09	44.95	47.63
LargeKitchenAppliances	375	375	3	720	*26.40*	*26.13*	*18.40*	43.20	45.60
MedicalImages	381	760	10	99	*26.32*	*25.26*	*24.87*	31.71	34.61
MiddlePhalanxOutlineAgeGroup	400	154	3	80	50.00	50.00	54.55	48.70	55.19
MiddlePhalanxOutlineCorrect	600	291	2	80	**30.24**	**30.24**	**27.84**	17.18	21.31
MiddlePhalanxTW	399	154	6	80	49.35	49.35	48.05	50.00	45.45
NonInvasiveFetalECGThorax1	1800	1965	42	750	20.97	*18.47*	*19.80*	22.14	**49.26**
NonInvasiveFetalECGThorax2	1800	1965	42	750	*13.54*	*12.26*	*11.45*	18.73	*15.78*
OSULeaf	200	242	6	427	**40.50**	**43.39**	**35.12**	25.21	*17.77*
PhalangesOutlinesCorrect	1800	858	2	80	**27.16**	**27.16**	**28.09**	21.68	19.46
Plane	105	105	7	144	0.00	0.00	0.00	0.00	0.00
ProximalPhalanxOutlineAgeGroup	400	205	3	80	19.51	19.51	20.98	17.56	19.02
ProximalPhalanxOutlineCorrect	600	291	2	80	**21.65**	**20.96**	**23.02**	8.59	9.97
ProximalPhalanxTW	400	205	6	80	24.39	24.39	25.37	24.39	22.44
RefrigerationDevices	375	375	3	720	**54.67**	**57.87**	**54.13**	44.27	**67.20**
ScreenType	375	375	3	720	60.80	64.27	65.60	64.27	59.20
ShapesAll	600	600	60	512	**23.00**	**23.67**	**17.33**	13.67	13.67
SmallKitchenAppliances	375	375	3	720	*31.73*	41.33	*34.67*	45.07	48.53
Strawberry	613	370	2	235	5.95	5.95	5.68	5.14	4.32
SwedishLeaf	500	625	15	128	**20.80**	**15.20**	**17.44**	7.36	**9.76**
SyntheticControl	300	300	6	60	0.67	0.67	0.67	0.33	0.00
Trace	100	100	4	275	0.00	5.00	1.00	0.00	0.00
TwoPatterns	1000	4000	4	128	0.00	**0.68**	0.00	0.00	0.00
UWaveGestureLibraryAll	896	3582	8	945	*5.28*	*4.75*	*4.16*	9.24	**88.11**
UWaveGestureLibraryX	896	3582	8	315	**26.91**	**24.46**	22.59	22.03	**24.23**
UWaveGestureLibraryY	896	3582	8	315	**36.21**	31.88	*29.93*	32.94	33.56
UWaveGestureLibraryZ	896	3582	8	315	**34.06**	**32.80**	**32.89**	29.09	**34.34**
Wafer	1000	6164	2	152	**2.01**	**0.44**	**0.71**	0.18	0.23
WordSynonyms	267	638	25	270	34.64	29.47	*23.35*	32.76	**43.10**
Worms	181	77	5	900	49.35	46.75	46.75	37.66	**57.14**
WormsTwoClass	181	77	2	900	42.86	41.56	35.06	38.96	48.05
Yoga	300	3000	2	426	16.37	15.67	*14.07*	17.10	**42.87**
All					**23.58**	**21.69**	**21.39**	20.21	**26.96**
Average					27.88	27.21	25.58	23.71	27.66
Wins					5	5	15	23	15



(a) DTW vs. ours

(b) ours w/o vs. w/ pre-training

Figure 9: Error rate comparison. Each point corresponds to one of the 52 datasets.

We used the 1-Nearest Neighbor (1-NN) rule as the classifier. More specifically, we used the distance given by the proposed method and then compared each test sample with all training samples. The class of the training sample with the minimum distance was considered as the classification result. We used the same 1-NN classification approach for the comparative methods.

4.2.3. Quantitative evaluation results

Table 2 shows classification error rates by the proposed method (ours), and the comparative methods, i.e., DTW, w-DTW, s-DTW, on 52 UCR2015 datasets[3]. As an ablation study, the performance of the proposed method without pre-training phase is also listed in this table. The error rates in red and blue indicate the least rate and the second least rate, respectively.

For many datasets, the proposed deep attentive time warping achieved lower error rates than the traditional DTW methods. This fact is confirmed by the number of wins; the proposed method shows the lowest error rates for 23 among 52 datasets.

Fig. 9(a) shows a pair-wise comparison between DTW and the proposed method. Each point corresponds to one of the 52 datasets. The 36 points below

[3] In UCR2018 [36], we found six datasets that satisfy the same conditions as UCR2015. The experimental evaluation results on these datasets are given in Appendix A.

the diagonal line indicate that the proposed method achieved a lower error rate than DTW for those datasets. Consequently, this figure also demonstrates a higher effectiveness of the proposed method.

As an ablation study, we observed the performance change by removing the pre-training phase. The accuracies of the proposed method without pre-training are shown in the rightmost column of Table 2 (“w/o pre-train.”) and summarized in Fig. 9(b) as a pairwise comparison with the method with pre-training. These results show that the positive effect of pre-training is confirmed on 36 datasets among the 52.

In order to make our evaluation more reliable, we conducted the McNemar’s test between the proposed method and each comparative method. The test results are shown in Table 2. If an error rate by a comparative method is printed in **bold**, the proposed method was superior to the comparative method with statistical significance at the 5% level by McNemar’s test. If *italic*, the proposed method is inferior with significance at the 5% level.

From the results of McNemar’s test, we can confirm the superiority of the proposed method over the comparative methods. More specifically, among 52 datasets, the proposed method was superior to DTW, w-DTW, and s-DTW, and w/o pre-training with the statistical significance at the 5% level on, 20, 22, 18, and 11 datasets, respectively. We can also see that inferior cases (italic) are much less than superior cases (bold). The row “All” in Table 2 shows the error rates of all test samples in all 52 datasets. McNemar’s test results in the “All” row also shows that the proposed method was superior to all the comparative methods at the 5% level — Precisely speaking, the proposed method was superior even at the 1% level.

In order to understand the characteristics of the proposed method, we analyzed the relationship between several dataset features (e.g., dataset size, time length, etc.) and win-lost cases. Among these features, time length shows the most evident relationship, as shown in Fig. 10. This figure shows the histograms of win cases and lost cases with respect to sample time length. To emphasize

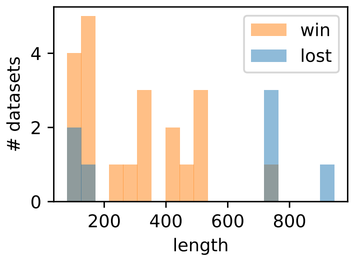


Figure 10: Histogram of win cases and lost cases with respect to time length. Note that we only picked up the datasets that show statistical significance at the 5% level by McNemar's test in Table 2 for emphasizing the difference between win cases and lost cases.

the difference between win cases and lost cases, we picked up the datasets that show statistical significance at the 5% level by McNemar's test in Table 2.

The histogram suggests that the lost cases are found for the datasets with very short or very long time lengths. A possible reason for this phenomenon is the fixed network architecture of the bipartite attention module. For example, for longer samples, the network is too shallow to exchange the information between their beginning and ending parts. In future work, we can try to use different network architectures according to the time length of samples.

We further compare the proposed method to results reported in literature. We collected results that propose distance measures for a 1-NN classifier, similar to the proposed method. The comparative methods use a wide variety of distance measure mechanisms, including derivative based methods, Complexity-Invariant Distance (CID) [37], Derivative Transform Distance (DTD$_C$) [38], and DTW Derivative Distance (DD$_{DTW}$) [39], dictionary distance based methods, Bag of Patterns (BOP) [40] and Bag of Symbolic

Table 3 lists the methods and the number times the proposed method has a higher accuracy over the comparison method (Wins), the number of times it had a lower accuracy (Loses), the number of ties, and the total number of datasets used in the comparisons. Note, each comparison method reports their results on different datasets within the 2015 UCR Time Series Archive. Therefore, we only count the datasets that are available. Also, since we limit the

Table 3: Comparison between proposed method and comparative methods on the 2015 UCR Time Series Archive datasets. The total number of datasets for each method is determined by the intersection of the datasets used by the proposed method and reported by the comparison methods.

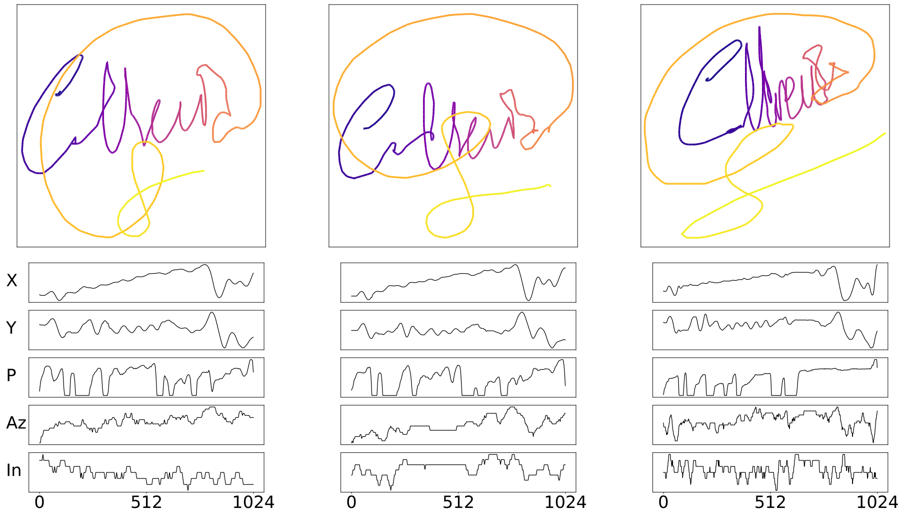
	Proposed Method			
Method	Wins	Losses	Ties	Total
BOP [40]	**8**	2	0	10
BOSS [41]	**12**	11	0	23
CID [37]	**13**	8	0	21
DD_{DTW} [39]	**8**	3	0	11
DTD_C [38]	9	**13**	0	22
MSM [42]	**8**	3	0	11
TWED [21]	**9**	2	0	11
WDTW [20]	**7**	2	0	9

proposed method to datasets with more than 100 training patterns, we exclude the reported datasets with less than 100 training patterns.

For most of the comparison methods, the proposed method performed much better. For BOSS and CID, the proposed method only had a small advantage and for DTD_C, the proposed method had fewer wins. This demonstrates that the proposed method is not only effective at providing an effective warping method, but also a robust distance measure for classification.

5. Experiments in Plug-in Scenario

We further conducted the experimental evaluation of the proposed deep attentive time warping in the plug-in scenario of Fig. 2 (b). The aims of this experiment are twofold. First, we evaluate the proposed method in a more practical task that needs representation learning in addition to time warping.



(a) Reference signature (b) Genuine signature (c) Skilled forgery signature

Figure 11: Examples of online signatures in MCYT-100.

Second, we compare the performance of the proposed method with state-of-the-art learnable time warping methods for the task.

We focused on online signature verification, which is a task to decide whether a test signature is a genuine signature or a skilled forgery (imitated by a forger). The reasons for using this task are as follows. The first and the most important reason is that several learnable time warping methods have been applied to the same public dataset, MYCT-100. As far as the authors know, there are no other common tasks to which various learnable time warping methods are applied. Second, this task requires representation learning; recent performance improvements on online signature verification owe to representation learning. Third, this task still needs further improvement; even though recent methods achieve 1.0% equal error rate (EER), verification error should be further minimized because of its expected reliability.

5.1. Online Signature Dataset

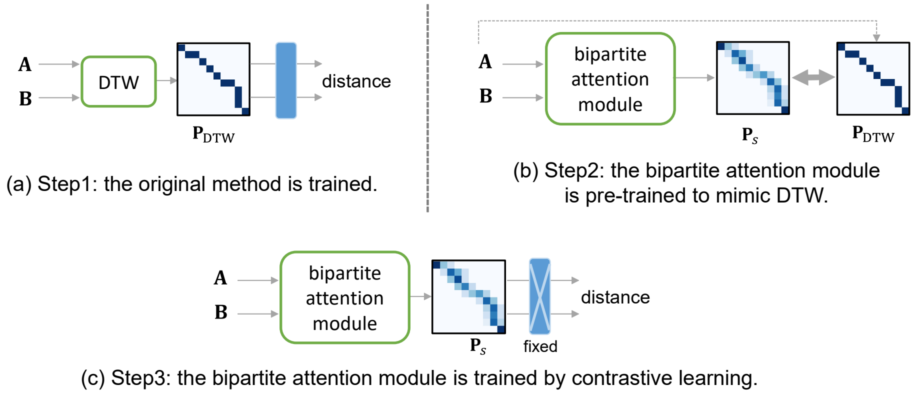
As one of the most common datasets for online signature verification, we use MCYT-100 dataset [43]. This dataset contains 100 subjects, each having 25 genuine signatures and 25 skilled forgeries. Fig. 11 shows several examples from the dataset. Each signature has five channels: 2D coordinates, pressure, azimuth, and altitude angles of the pen tip. We resized the temporal length to 1,024 by following the experimental setup of the comparative methods (such as PSN and DDTW).

According to the tradition of the task, we conduct multiple experiments under different train-test ratios. Specifically, the first $\eta\%$ of subjects ($\eta \in \{50, 60, 70, 80, 90\}$) were used for training and the remaining subjects for testing. For testing, the first five genuine signatures of each subject in the test set were used as reference signatures. The remaining genuine signatures and all the skilled forgeries were used as test signatures. For each test signature, its distances to the corresponding five reference signatures were averaged. Based on this averaged distance, all test signatures of all subjects were sorted to form a ranking list. EER was finally calculated as the traditional evaluation metric of the task.

5.2. Comparative methods

In the plug-in scenario experiment, we used two elementary methods and three state-of-the-art methods. The former methods are DTW [2] and simple Siamese Network (Siamese). DTW takes either raw signatures or handcrafted features [44] as input. Siamese was learned with either a global contrastive loss [32] or a local embedding loss [23].

The state-of-the-arts methods are Prewarping Siamese Network (PSN) [23], Time-Aligned Recurrent Neural Networks (TARNN) [24], and Deep DTW (DDTW) [25]. In the latter method, DTW is embedded in a metric learning framework to realize learnable time warping. More specifically, the original PSN and TARNN use DTW before Siamese networks, whereas the original DDTW af-


A
B
DTW
$\mathbf{P}_{\text{DTW}}$
distance
A
B
bipartite attention module
$\mathbf{P}_s$
$\mathbf{P}_{\text{DTW}}$
A
B
bipartite attention module
$\mathbf{P}_s$
fixed
distance

(a) Step1: the original method is trained.

(b) Step2: the bipartite attention module is pre-trained to mimic DTW.

(c) Step3: the bipartite attention module is trained by contrastive learning.

Figure 12: Three-step training process for the plug-in scenario (for PSN/TARNN). Each blue box is the Siamese network for contrastive representation learning. Note that for DDTW, the Siamese network is placed *before* the bipartite attention module.

ter. As the Siamese networks, PSN and DDTW employ CNN, whereas TARNN employs RNN.

For the comparison, we plugged the proposed method in the above methods and observed how the performance changed. Specifically, we replace the DTW module in PSN, DDTW, and TARNN by the proposed method and then train the entire network by the process of 5.3.

5.3. Implementation details

Fig. 12 shows the three-step training process for the deep attentive time warping plugged in PSN, TARNN, and DDTW. First, the original model with the standard DTW is trained with its original loss functions. (The details can be found in the original papers [23–25].) Second, the bipartite attention module is pre-trained with the loss (3), for providing a similar attention weight matrix to the DTW result. Finally, the bipartite attention module is trained with the entire network, while keeping the weights of the Siamese network. It is theoretically possible to train the entire network in an end-to-end manner. However, our preliminary trials proved that this three-step process gives more stable results.

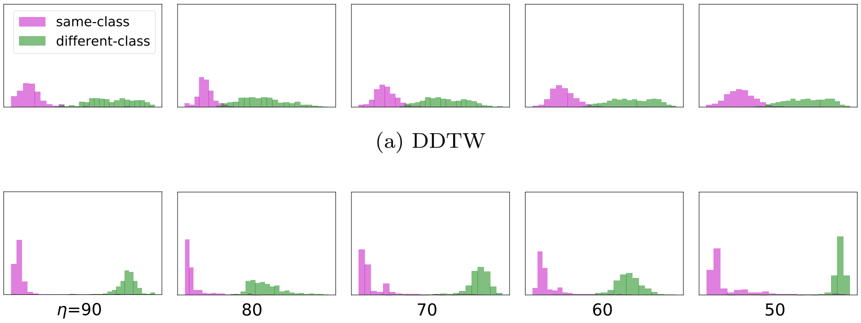
Table 4: EERs (%) of online signature verification on MCYT-100. EER in red and blue indicate the least and the second least rates, respectively.

	Percentage (η%) of Training Data				
Method	90	80	70	60	50
DTW [2, 44]	4.00	3.00	4.17	4.37	4.60
w/ raw signatures	5.00	6.25	5.73	6.37	6.96
Siamese	5.50	6.80	6.27	7.33	8.40
w/ local embedding loss [23]	3.50	3.40	3.75	3.75	5.50
PSN [23]	1.50	2.25	3.17	2.75	3.00
+ ours (plug-in)	1.00	1.75	2.33	2.13	2.70
w/o pre-training	1.00	2.50	3.67	3.50	4.10
TARNN [24]	1.00	3.00	3.50	4.25	4.50
+ ours (plug-in)	0.50	2.25	2.67	2.88	2.80
w/o pre-training	1.50	2.50	3.17	3.25	5.00
DDTW [25]	1.00	2.20	2.53	2.25	2.40
+ ours (plug-in)	0.50	2.00	2.33	2.13	2.20
w/o pre-training	1.50	4.00	4.83	3.50	3.90

The network architecture of the bipartite attention module is the same as the stand-alone scenario. The learning rate was chosen from 0.1, 0.01, and 0.001 in Steps 1 and 3, and set to 0.001 in Step 2. The hyper-parameter τ in the contrastive loss was set to 1.4 by using the validation set. Adam was used as the optimizer. The training was conducted up to $10,000$ iterations in Steps 1 and 3, and up to $1,000$ iterations in Step 2. The batch size was set to 30, where 10 are the same-class pairs and 20 are the different-class pairs).

5.4. Results

Table 4 shows the EERs on MCYT-100. The accuracy of the state-of-the-art methods, i.e., PSN, TARNN, and DDTW, has all been improved by replacing



(a) DDTW

(b) ours

Figure 13: Distance histograms by the proposed method and DDTW [25] on MCYT-100. The horizontal axis is the distance and the vertical axis is normalized.

DTW with the proposed method. This proves that the proposed method is consistently effective as a plug-in to existing learnable time warping frameworks. In addition, from the comparison between PSN and DDTW, the plug-in location (i.e., before or after the representation learning module) is not very important.

Removing pre-training from the proposed method degrades the performance significantly. This fact confirms the necessity of the proposed pre-training method in improving discriminative power and stabilizing the inference of bipartite attention matrices.

Fig. 13 shows the distance histograms of the same-class and different-class pairs by the proposed method and DDTW. The proposed method shows a much smaller overlap than DDTW. DDTW focuses only on contrastive representation learning; in contrast, the proposed method trains the attention weight matrix (i.e., soft-correspondence) in contrastive learning, in addition to representation learning. This makes the proposed method more discriminative than DDTW.

6. Conclusion

In this paper, we proposed a novel neural network-based time warping method, called deep attentive time warping. The proposed method is based on a new attention module, called the bipartite attention module, between two time series inputs. The module is trained by contrastive metric learning to achieve a learnable and task-adaptive time warping and to improve the trade-off between robustness against time distortion and discriminative power. The effectiveness of the proposed method was confirmed through two scenarios. The first was a stand-alone scenario, where the proposed method was used as a learnable time warping method and compared with the standard DTW and other time warping methods. Through qualitative and quantitative evaluations with Unipen and UCR datasets, the expected effectiveness was confirmed. The second was a plug-in scenario, where the proposed method is embedded in neural network-based metric learning frameworks with representation learning. Through a comparative study with state-of-the-art learnable time warping methods, the effectiveness of the proposed method was further confirmed.

The limitations of this paper are as follows. First, in the current framework, the regulation of the warping flexibility relies on the pre-training to mimic the standard DTW and the soft constraints implicitly imposed by the trained bipartite attention module; this means no explicit penalty for the violation of several reasonable regulations, such as monotonicity and continuity of warping. Although we confirmed the performance superiority over the standard DTW with those warping regulations, there is still a possibility that the introduction of some explicit regulations will further improve the performance. Second, we optimize the network architectures according to the characteristics of the dataset. In this paper, we used the same architecture for all UCR2015 datasets, and therefore the performance degrades for too long or too short time-series samples, as revealed by the analysis in Section 4.2.3. From a practical viewpoint, architecture optimization for better performance will be an important future work. Third,

we have not directly utilized the soft-correspondence (represented by the attention weight matrix) in the final distance evaluation. In fact, the attention weight matrix can be seen as a novel feature showing the relationship between two time series, and therefore we can extract some useful features from it for final distance evaluation and/or final decision making.

Acknowledgments

This work was partially supported by MEXT-Japan (Grant No. J17H06100 and J22H00540).

References

[1] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, P. Muller, Deep learning for time series classification: A review, Data Mining and Knowledge Discovery 33 (4) (2019) 917–963.

[2] H. Sakoe, S. Chiba, Dynamic programming algorithm optimization for spoken word recognition, IEEE Transactions on Acoustics, Speech and Signal Processing 26 (1) (1978) 43–49.

[3] C. A. Ratanamahatana, E. J. Keogh, Three myths about dynamic time warping data mining, in: SIAM International Conference on Data Mining (SDM), 2005, pp. 506–510.

[4] J. Lines, A. J. Bagnall, Time series classification with ensembles of elastic distance measures, Data Mining and Knowledge Discovery 29 (3) (2015) 565–592.

[5] E. Hoffer, N. Ailon, Deep metric learning using triplet network, in: International Conference on Learning Representations (ICLR) Workshop, 2015.

[6] W. Chen, X. Chen, J. Zhang, K. Huang, Beyond triplet loss: A deep quadruplet network for person re-identification, in: Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1320–1329.

[7] H. O. Song, S. Jegelka, V. Rathod, K. Murphy, Deep metric learning via facility location, in: Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2206–2214.

[8] M. Kaya, H. S. Bilge, Deep metric learning: A survey, Symmetry 11 (2019) 1066.

[9] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, L. Song, SphereFace: deep hypersphere embedding for face recognition, in: Computer Vision and Pattern Recognition (CVPR), 2017, pp. 6738–6746.

[10] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, W. Liu, CosFace: large margin cosine loss for deep face recognition, in: Computer Vision and Pattern Recognition (CVPR), 2018, pp. 5265–5274.

[11] J. Deng, J. Guo, N. Xue, S. Zafeiriou, ArcFace: additive angular margin loss for deep face recognition, in: Computer Vision and Pattern Recognition (CVPR), 2019, pp. 4690–4699.

[12] J. Mueller, A. Thyagarajan, Siamese recurrent architectures for learning sentence similarity, in: AAAI Conference on Artificial Intelligence (AAAI), 2016, pp. 2786–2792.

[13] H. Coskun, D. J. Tan, S. Conjeti, N. Navab, F. Tombari, Human motion analysis with deep metric learning, in: European Conference on Computer Vision (ECCV), 2018, pp. 693–710.

[14] D. Roy, C. K. Mohan, K. S. R. Murty, Action recognition based on discriminative embedding of actions using Siamese networks, in: IEEE International Conference on Image Processing (ICIP), 2018, pp. 3473–3477.

[15] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, R. Shah, Signature verification using a Siamese time delay neural network, in: Conference on Neural Information Processing Systems (NeurIPS), 1993, pp. 737–744.

[16] R. Tolosana, R. Vera-Rodríguez, J. Fiérrez, J. Ortega-Garcia, Exploring recurrent neural networks for on-line handwritten signature biometrics, IEEE Access 6 (2018) 5128–5138.

[17] K. Ahrabian, B. BabaAli, Usage of autoencoders and Siamese networks for online handwritten signature verification, Neural Computing and Applications 31 (12) (2019) 9321–9334.

[18] I. Guyon, L. Schomaker, R. Plamondon, M. Liberman, S. Janet, UNIPEN project of on-line data exchange and recognizer benchmarks, in: International Conference on Pattern Recognition (ICPR), 1994, pp. 29–33.

[19] Y. Chen, E. Keogh, B. Hu, N. Begum, A. Bagnall, A. Mueen, G. Batista, The UCR time series classification archive, `www.cs.ucr.edu/~eamonn/time_series_data/` (July 2015).

[20] Y. Jeong, M. K. Jeong, O. A. Omitaomu, Weighted dynamic time warping for time series classification, Pattern Recognition 44 (9) (2011) 2231–2240.

[21] P. Marteau, Time warp edit distance with stiffness adjustment for time series matching, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (2) (2009) 306–318.

[22] M. Cuturi, M. Blondel, Soft-DTW: A differentiable loss function for time-series, in: International Conference on Machine Learning (ICML), 2017, pp. 894–903.

[23] X. Wu, A. Kimura, S. Uchida, K. Kashino, Prewarping Siamese network: Learning local representations for online signature verification, in: IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2019, pp. 2467–2471.

[24] R. Tolosana, R. Vera-Rodríguez, J. Fiérrez, J. Ortega-Garcia, DeepSign: Deep on-line signature verification, IEEE Transactions on Biometrics, Behavior, and Identity Science 3 (2) (2021) 229–239.

[25] X. Wu, A. Kimura, B. K. Iwana, S. Uchida, K. Kashino, Deep dynamic time warping: End-to-end local representation learning for online signature verification, in: IEEE International Conference on Document Analysis and Recognition (ICDAR), 2019, pp. 1103–1110.

[26] Z. Che, X. He, K. Xu, Y. Liu, DECADE: A deep metric learning model for multivariate time series, in: Workshop on Mining and Learning from Time Series, 2017.

[27] J. Grabocka, L. Schmidt-Thieme, NeuralWarp: Time-series similarity with warping networks (2018). arXiv:1812.08306.

[28] S. Matsuo, X. Wu, G. Atarsaikhan, A. Kimura, K. Kashino, B. K. Iwana, S. Uchida, Attention to warp: Deep metric learning for multivariate time series, in: IEEE International Conference on Document Analysis and Recognition (ICDAR), 2021, pp. 350–365.

[29] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, in: International Conference on Learning Representations (ICLR), 2015.

[30] T. Luong, H. Pham, C. D. Manning, Effective approaches to attention-based neural machine translation, in: Conference on Empirical Methods in Natural Language Processing (EMNLP), 2015, pp. 1412–1421.

[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Conference on Neural Information Processing Systems (NeurIPS), 2017, pp. 5998–6008.

[32] R. Hadsell, S. Chopra, Y. LeCun, Dimensionality reduction by learning an invariant mapping, in: Computer Vision and Pattern Recognition (CVPR), 2006, pp. 1735–1742.

[33] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), 2015, pp. 234–241.

[34] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: International Conference on Learning Representations (ICLR), 2015.

[35] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification, in: IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1026–1034.

[36] H. A. Dau, A. Bagnall, K. Kamgar, C.-C. M. Yeh, Y. Zhu, S. Gharghabi, C. A. Ratanamahatana, E. Keogh, The ucr time series classification archive, `https://www.cs.ucr.edu/~eamonn/time_series_data_2018/` (October 2018).

[37] G. E. A. P. A. Batista, X. Wang, E. J. Keogh, A complexity-invariant distance measure for time series, in: SIAM nternational Conference on Data Mining (SDM), 2011, pp. 699–710.

[38] T. Górecki, M. Luczak, Non-isometric transforms in time series classification using DTW, Knowledge Based Systems 61 (2014) 98–108.

[39] T. Górecki, M. Luczak, Using derivatives in time series classification, Data Mining and Knowledge Discovery 26 (2) (2013) 310–331.

[40] J. Lin, R. Khade, Y. Li, Rotation-invariant similarity in time series using bag-of-patterns representation, Journal of Intelligent Information Systems 39 (2) (2012) 287–315.

[41] P. Schäfer, The BOSS is concerned with time series classification in the presence of noise, Data Mining and Knowledge Discovery 29 (6) (2015) 1505–1530.

[42] A. Stefan, V. Athitsos, G. Das, The move-split-merge metric for time series, IEEE Transactions on Knowledge and Data Engineering 25 (6) (2013) 1425–1438.

[43] J. Ortega-Garcia, et al., MCYT baseline corpus: A bimodal biometric database, IEE Proceedings – Vision, Image and Signal Processing 150 (6) (2003) 395–401.

[44] M. Martinez-Diaz, J. Fiérrez, R. P. Krish, J. Galbally, Mobile signature verification: Feature robustness and performance comparison, IET Biometrics 3 (4) (2014) 267–277.

Table 5: Error rate (%) on additional six datasets from UCR2018. Error rates in red and blue indicate the least and the second least rates, respectively.

Dataset	Train	Test	Class	Length	DTW	w-DTW	s-DTW	ours	w/o pre-train
Crop	7200	16800	24	46	33.00	28.83	31.46	37.13	32.13
FreezerRegularTrain	150	2850	2	301	10.00	9.30	6.84	0.00	0.00
GunPointAgeSpan	135	316	2	150	8.00	3.48	1.58	0.00	0.00
GunPointMaleVersusFemale	135	316	2	150	0.00	2.53	1.58	0.00	0.00
GunPointOldVersusYoung	136	315	2	150	16.00	3.49	0.00	0.00	0.00
PowerCons	180	180	2	144	12.00	7.78	9.44	0.00	0.00

Appendix A Result on datasets from UCR2018

In addition to results of UCR2015 in Table 2, we conducted the same experimental evaluation for UCR2018 [36]. In UCR2018, we found six datasets that satisfy the same conditions (the sample size, the time length, and the fixed-length samples) as the UCR2015 experiment. Table 5 shows the error rate of the datasets. Except for “Crop,” which contains many training samples beneficial for 1NN classification by DTW, ours could achieve the best accuracy (even perfect).