Randomized Time Warping for Motion Recognition

Chendra Hadi Suryanto[a,*], Jing-Hao Xue[b], Kazuhiro Fukui[a]

[a]*Department of Computer Science, Graduate School of Systems and Information Engineering, University of Tsukuba, 1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan*
[b]*Department of Statistical Science, University College London, London, WC1E 6BT, United Kingdom*

Abstract

Dynamic time warping (DTW) has been widely used for the alignment and comparison of two sequential patterns. In DTW, dynamic programming is used to avoid an exhaustive search for the alignment. In this paper, we propose a randomized extension of the DTW concept, termed randomized time warping (RTW), for motion recognition. RTW generates time elastic (TE) features by randomly sampling the sequential data while retaining the temporal information. A set of TE features is represented by a low-dimensional subspace, called the sequence hypothesis (Hypo) subspace, and the similarity between two sequential patterns is defined by the canonical angles between the two corresponding Hypo subspaces. In essence, RTW simultaneously computes multiple degrees of similarities between a number of warped patterns' pair candidates, while in practice, RTW generalizes the Hankel matrix commonly used in modeling of system dynamics. We demonstrate the applicability of RTW through experiments on gesture recognition using three public datasets, namely, the Cambridge gesture database, a subset of the one-shot-learning dataset from the ChaLearn Gesture Challenge, and the KTH action dataset.

Keywords: feature extraction, dynamic time warping, subspace method, Hankel matrix, motion recognition

1. Introduction

Dynamic time warping (DTW), which is also termed dynamic programming-matching, has been widely used for sequential data analysis. Early uses of DTW range from the comparison of amino acids sequences in bioinformatics [1], through speech recognition [2], to motion analysis [3]. The core idea of DTW is to search for the best alignment of two sequential patterns by optimizing a warping function, which specifies the sequential correspondence between them. Since the number of possible combinations of warped patterns is exponentially large, to avoid exhaustive search dynamic programming has been used, which can effectively optimize the alignment score and produce the alignment path of the most similar warped patterns.

Although DTW is a very useful and widely applicable tool for sequence analysis, it has several limitations when applied to tasks of classifying multiple sequences, such as gesture recognition with many kinds of hand shapes and personal identification by gait recognition. Here are the issues that we will address in this paper.

1. Since dynamic programming is basically a deterministic approach, the obtained solution is likely to be sub-optimal for the sequential data that contains large intra-variation in the temporal structure.
2. The alignment is typically done by trying to match an input sequence to each reference sequence in a given set. This can lead to a high computational cost when the number of the reference sequences to be considered is large.
3. DTW has no internal mechanism to remove or ignore irrelevant variation that may affect the classification result. For example, variation of lighting conditions in video data or speakers in speech data can significantly lower the performance of a classification method using DTW. That is, DTW-based classification methods are sensitive to these undesirable effects.

To tackle these issues, we generalize the notion of DTW to construct a new method for sequential data analysis, which is termed *randomized time warping* (RTW). The core idea of RTW is essentially to simultaneously search for the most similar warped patterns

*Corresponding author: Tel./Fax. +81-29-853-5544.
Email addresses: chendra@cvlab.cs.tsukuba.ac.jp (Chendra Hadi Suryanto), jinghao.xue@ucl.ac.uk (Jing-Hao Xue), kfukui@cs.tsukuba.ac.jp (Kazuhiro Fukui)

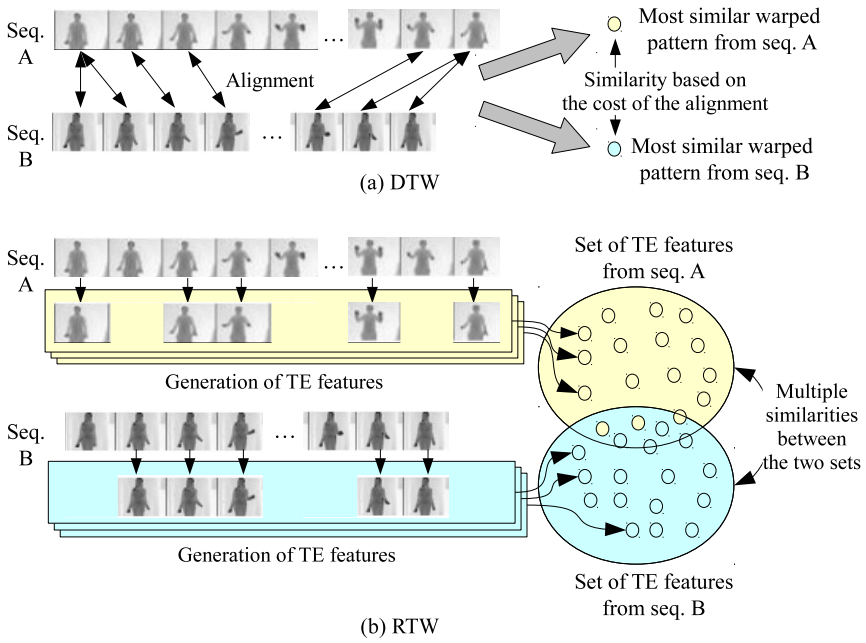

Figure 1: Comparison between DTW and RTW. (a) DTW searches for the most optimal alignment in a large space through dynamic programming. The outputs of DTW are the most similar warped patterns and the cost of the alignment. (b) RTW generates many candidate warped patterns, called time elastic (TE) features, and then compares the sets of the candidates. The outputs of RTW are multiples of the highest similarities between the two sets.

from a number of candidates which are prepared beforehand through randomization. Figure 1 illustrates the difference between DTW and our RTW approach.

Instead of searching for the most similar warped patterns using dynamic programming, RTW progressively generates a set of time warped patterns, called *time elastic* (TE) features, through repeated random subsampling while preserving the original temporal order. We utilize this bagging-like strategy to ensure that the set of the TE features contains sufficient discriminative frames with high probability. The use of TE features converts the comparison of two sequences to the comparison of two sets of TE features. Figure 2 shows the comparison process between two sets of the TE features. The comparison is conducted using a subspace-based method, in which each set of TE features is represented as a low-dimensional subspace, called a sequence *hypothesis* (Hypo) subspace. Finally, the similarity between the two sequences is defined by the average of multiple canonical angles θ_i between the two Hypo subspaces. We regard the canonical vectors that form the canonical angles as pseudo-warped patterns (Further discussion is provided in Section 3.2). This approach can provide a promising solution to each of the DTW issues previously mentioned:

1. Since random sampling is able to generate a large number of time warped patterns (TE features), our RTW approach is non-deterministic and can deal with a huge number of possible combinations of warped patterns with various time-scales, and thus is able to tackle the issue with large intra-variation in the temporal structure.
2. Since our approach uses the compact subspace-representation, exhaustive matching between all possible TE features is avoided. Each Hypo subspace can contain the TE features from multiple sequences and the canonical angles between two subspaces can be calculated with simple linear algebra. Hence RTW can alleviate the issue of high computational costs.
3. Our approach is based on a subspace method, which can remove or reduce the undesirable effects of irrelevant features. This enables RTW to mitigate the third issue and thus improve the performance of classification

To demonstrate the effectiveness of our approach, we focus on gesture recognition in this paper. We conducted experiments on gesture recognition using three public datasets, namely, the Cambridge hand gesture dataset [4] which contains variations of lighting con-

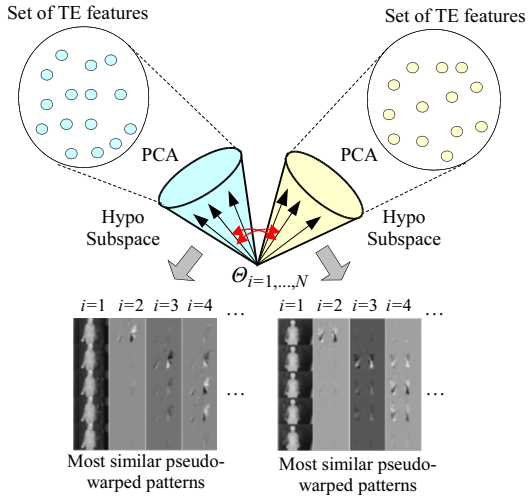

Figure 2: The comparison process for two sets of TE features in RTW.

dition, a subset of ChaLearn gesture dataset [5] which contains very limited training samples, and the KTH action dataset [6] which is a widely used benchmark dataset for action recognition. In addition, we also demonstrate the extensibility of RTW by including a subspace learning method using Grassmann discriminant analysis [7].

In the next section, we start with a review of subspace methods for gesture recognition, which is followed by a review of related works on DTW. Then we describe how RTW tackles the DTW issues and the relationship to the Hankel matrix in Section 3. The classification framework of RTW is provided in Section 4. An adaptation of Grassmann discriminant analysis in the classification framework is discussed in Section 5. Experimental results are reported in Section 6. Finally, conclusions and indication of future work are given in Section 7.

2. Related work

Although we regard the concept of RTW as a generalization of DTW, the practical process of RTW is partly related to other types of methods such as subspace methods and the methods based on canonical correlation analysis (CCA). In the following, we first review such related methods, including the extensions of the original DTW. Then we describe several DTW-based methods for gesture recognition.

In the usage of CCA-based methods, to encode the space-time volume of a gesture, [4] used a third-order tensor-based CCA with AdaBoost for feature selection. In [8], the tensor was factorized into a set of tangent spaces, to which the classification of the video was applied. In [9], two types of subspaces representing activity motion were developed, one from the images of a sequence and the other from linear autoregressive-moving-average models. Then the classification was done on a Grassmann manifold, on which each subspace was interpreted as a point. In [10], dynamic systems of motions were modeled by Hankel matrices of extracted features. Then subspaces spanning the columns of the Hankel matrices were obtained by using discriminant canonical correlation [11], and finally support vector machines were used for classification. However, these subspace-based methods can suffer when there is only a small number of training samples.

In the development of DTW, stochastic DTW was proposed in [12] to tackle the intra-variation problem in speech recognition. In stochastic DTW, the distances and path costs of conventional DTW were replaced with conditional probabilities and transition probabilities, respectively. Stochastic DTW shows that the DTW method is strongly related to the hidden Markov model (HMM) approach [12]. In sequential data analysis throughout the years, HMMs have been favored over DTW due to their better generalization to sets of samples, in which exhaustive pair-wise comparison can be avoided [13]. This also led to the development of statistical DTW, which is equivalent to the HMM approach and generates a statistical model from the set of samples [14, 13]. However, HMMs require many assumptions in generalizing the system dynamics of time-series data [10]. To avoid these difficulties, the Hankel matrix was used to approximate HMMs, especially in system identification tasks [15, 16]. In Section 3.3 we discuss how, in the implementation of RTW, the matrix of TE features can be regarded as a generalization of the Hankel matrix.

Applications of DTW to gesture recognition have been reviewed in several survey papers [17, 18, 19]. Recent extensions of DTW include the method in [20], called Isotonic CCA, which generalized the concept of DTW by imposing a monotonicity constraint on CCA. Canonical time warping (CTW) in [21] combined DTW with CCA to take spatial variability into account in the alignment process. In [22] DTW was extended to generalized time warping (GTW), which used multiple CCA to find an optimal nonlinear temporal transformation and a low-dimensional space embedding of multiple multi-modal sequences. The ideas behind [20, 21, 22] may be used to enhance the performance of DTW. However, as in classical DTW, the computational cost increases rapidly with the number of reference sequences to be compared.

3. Randomized time warping

First, we demonstrate that TE features have valid statistical properties as a key component of the framework of RTW in Section 3.1. Then we describe how to simultaneously compute multiple similarities between two sets of TE features in Section 3.2. Finally we discuss the relationship of the matrix from the set of the TE features with the Hankel matrix in Section 3.3.

3.1. Statistical properties of time elastic features

To deal with large intra-variation of temporal structure, we require features to cover both the local and global information of the temporal structure: global information accommodates the overall temporal structure; local information deals with fragments of the temporal structure.

In the following, we show that a set of TE features has such favorable properties. We consider a set of image sequences each of which consists of a number of ordered images. Nevertheless, the following discussion can easily be generalized to other types of data.

Let $\{\mathbf{x}_1, \ldots, \mathbf{x}_{N^{(s)}}\}$ be the ordered data of sequence s, where $N^{(s)}$ is the length of the sequence. Let $\mathbf{x}_i \in \mathbb{R}^f$ be the original feature vector of an image. An $f \times n$-dimensional TE feature vector $\mathbf{s} = [\mathbf{y}_1^{\mathrm{T}}\ \mathbf{y}_2^{\mathrm{T}}\ \ldots\ \mathbf{y}_n^{\mathrm{T}}]^{\mathrm{T}}$ is created by randomly selecting n images from a sequence s, such that $\mathbf{y}_1, \ldots, \mathbf{y}_n \in \{\mathbf{x}_1, \ldots, \mathbf{x}_{N^{(s)}}\}$, $t(\mathbf{y}_1) < \ldots < t(\mathbf{y}_n)$, where $t(\cdot)$ denotes the original order of the image. The value of n, which denotes the number of image selected to construct a TE feature, also corresponds to the number of effective frames needed for recognition, which has been studied in [23].

In statistics, $t(\mathbf{y}_1)$ is the random variable for the minimal image order of the n images selected into $\mathbf{s}$, and $t(\mathbf{y}_n)$ is the maximal order. That is, $t(\mathbf{y}_j)$ is the jth order statistic for the TE features and is in the set of $\{j, \ldots, N^{(s)} - n + j\}$. Over this support, the probability that $t(\mathbf{y}_j) = k$ can be written as

$$Pr(t(\mathbf{y}_j) = k) = \frac{\binom{k-1}{j-1}\binom{N^{(s)}-k}{n-j}}{\binom{N^{(s)}}{n}} . \qquad (1)$$

The probability mass functions for $t(\mathbf{y}_j)$, $j = 1, \ldots, n$, are shown in Figure 3 where $n = 5$ and $N^{(s)} = 10$. This describes a statistical mechanism for the extraction of TE features, applicable over the whole sequence rather than constrained to a local neighborhood. Images located near the edges of a motion are most likely to be selected as the start and end blocks of a TE feature. This indicates that we are able to collect the global structures of temporal information as well as local temporal

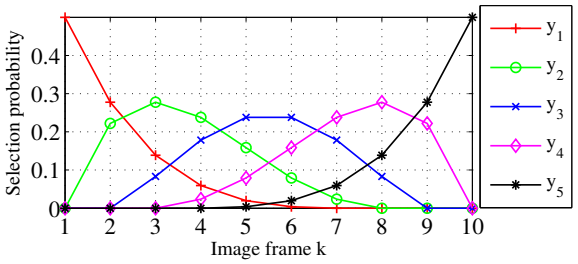

Figure 3: The probability for each image from a sequence of 10 images to be randomly selected into a TE feature vector $[\mathbf{y}_1^{\mathrm{T}}\ \mathbf{y}_2^{\mathrm{T}}\ \mathbf{y}_3^{\mathrm{T}}\ \mathbf{y}_4^{\mathrm{T}}\ \mathbf{y}_5^{\mathrm{T}}]^{\mathrm{T}}$.

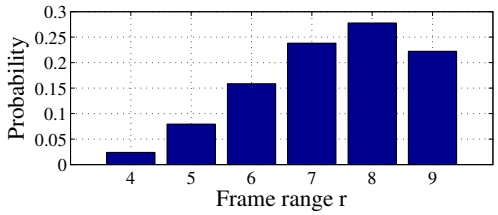

Figure 4: The probabilities of frame ranges for the selected images.

structures. The probability of a TE feature containing a frame range r, for $r = n - 1, \ldots, N^{(s)} - 1$, can be formulated as

$$Pr(t(\mathbf{y}_n) - t(\mathbf{y}_1) = r) = (N^{(s)} - r)\frac{\binom{r-1}{n-2}}{\binom{N^{(s)}}{n}} . \qquad (2)$$

As an illustrative example, Figure 4 shows the probability distribution of the frame range for the 5-block ($n = 5$) TE features generated from a motion containing 10 images ($N^{(s)} = 10$). The frame range indicates the extent of the globality of the temporal information encoded in the TE feature. For example, the TE features containing images ordered $1, 2, 3, 4, 5$ and $2, 3, 5, 9, 10$ have frame ranges of 4 and 8, respectively.

3.2. Simultaneous verification of multiple sequence hypotheses

Through the repetition of random sampling, we ensure that the set of the TE features contains sufficient discriminative frames with high probability. However, due to the randomness, not all the selected features in the set contain discriminative information. We reduce this redundancy by generating a subspace through applying principal component analysis (PCA) to the set of the TE features in RTW.

Let the procedure of random selection described in the previous subsection be repeated R times, such that

we obtain $\mathbf{s}_1, \ldots, \mathbf{s}_R$. Subsequently, a correlation-like matrix $\mathbf{A}$, which corresponds to the set of the TE feature vectors, can be computed as

$$\mathbf{A} = \frac{1}{R} \sum_{i=1}^{R} \mathbf{s}_i \mathbf{s}_i^{\mathrm{T}} \ . \tag{3}$$

We apply PCA to construct an N-dimensional subspace by computing the eigenvectors $[\boldsymbol{\phi}_1, ..., \boldsymbol{\phi}_N]$ of the matrix $\mathbf{A}$. A set of TE features generated from a sequence contains various possible warped patterns, each of which corresponds to one hypothesis. In this sense, the subspace generated from a set of TE features is called a sequence *hypothesis* (Hypo) subspace.

One advantage of using the Hypo subspace to represent the set of TE features is that we can deal with multiple sequences. In the case when there are multiple reference sequences that belong to the same class, it is possible to represent the set of their TE features together in one Hypo subspace. Thus, the recognition of an unknown sequence is more efficient, because it is not necessary to compare the unknown sequence to every reference sequence that belongs to the same class.

3.2.1. Computation of canonical angles

Next, we describe how to compute the similarities between two Hypo subspaces. The usage of canonical angles for similarity measures is also known as the mutual subspace method, a technique widely used in image-set-based 3D object recognition [24, 25, 26, 27]. Let $\mathcal{P}_c$ be an N-dimensional reference subspace of class c, and $\mathcal{Q}$ be an M-dimensional input subspace. The first canonical angle θ_1 is defined by

$$\cos\theta_1 = \max_{\boldsymbol{u}_i \in \mathcal{Q}} \max_{\boldsymbol{v}_i \in \mathcal{P}_c} \boldsymbol{u}_i^{\mathrm{T}} \boldsymbol{v}_i \ , \tag{4}$$

subject to $\boldsymbol{u}_i^{\mathrm{T}} \boldsymbol{u}_i = \boldsymbol{v}_i^{\mathrm{T}} \boldsymbol{v}_i = 1, \boldsymbol{u}_i^{\mathrm{T}} \boldsymbol{u}_j = \boldsymbol{v}_i^{\mathrm{T}} \boldsymbol{v}_j = 0, i \neq j$. A practical method of finding $\cos\theta_i$ ($i = 1, \ldots, M$ if $M \leq N$, and $0 \leq \theta_1 \leq \ldots \leq \theta_M \leq \frac{\pi}{2}$) is by computing the singular values of the matrix $\mathbf{W} = \mathbf{U}^{\mathrm{T}}\mathbf{V}$, where $\mathbf{U} = [\boldsymbol{\phi}_1, ..., \boldsymbol{\phi}_M]$, $\mathbf{V} = [\boldsymbol{\psi}_1, ..., \boldsymbol{\psi}_N]$, and $\boldsymbol{\phi}_i$ and $\boldsymbol{\psi}_i$ are the orthogonal basis vectors of the subspaces $\mathcal{Q}$ and $\mathcal{P}_c$, respectively.

3.2.2. Importance of multiple canonical angles

The similarities between two Hypo subspaces are defined by the cosines of the canonical angles θ_i. The first canonical angle θ_1 corresponds to the largest canonical correlation between the two sets of TE features, which can be interpreted as a distance between the two most similar warped patterns in the two corresponding Hypo subspaces. The second canonical angle θ_2 corresponds to the second largest canonical correlation between the two sets of TE features, and so on. The use of only the first canonical angle can lead to less stable recognition performance, as in a DTW approach that considers only the most similar warped patterns. This suggests that multiple canonical angles are required in order to take all possible warped patterns in the Hypo subspace into consideration and to achieve more stable performance. We use the average of the similarities of all canonical angles as the final similarity:

$$Sim(\mathcal{Q}, \mathcal{P}_c) = \frac{1}{M} \sum_{i=1}^{M} \cos^2\theta_i \ . \tag{5}$$

Since the multiple similarities between many warped patterns are computed at the same time, we regard RTW as essentially conducting multiple DTWs simultaneously.

Furthermore, a pair of canonical vectors $\hat{\boldsymbol{u}}_i$ and $\hat{\boldsymbol{v}}_i$ that form canonical angle θ_i can be obtained as follows:

$$\hat{\boldsymbol{u}}_i = \mathbf{U}\boldsymbol{y}_i, \quad \hat{\boldsymbol{v}}_i = \mathbf{V}\boldsymbol{z}_i, \tag{6}$$

where $\boldsymbol{y}_i$ and $\boldsymbol{z}_i$ are, respectively, the left and right singular vectors of $\mathbf{W}$. These canonical vectors can be regarded as the most similar *pseudo-time-warped patterns* generated through the linear combination of TE features.

3.3. Relationship to the Hankel matrix

Conceptually, RTW generalizes DTW. In its implementation, RTW uses a matrix of the set of TE features, which can also be regarded as a generalization of the Hankel matrix.

The Hankel matrix $\mathbf{H}$ is defined as a matrix in which the elements are skewed diagonally:

$$\mathbf{H}_{i,j} = \mathbf{H}_{i-1,j+1} \ , \tag{7}$$

where i and j are row and column indices, respectively. In this approach, a column of the Hankel matrix contains n blocks of the f-dimensional feature vector $\mathbf{x}_i$ from an image sequence, where $\mathbf{x}_i$ ($i = 1, ..., N^{(s)}$) indicates the feature vector of the ith image of sequence s and $N^{(s)}$ is the number of images in sequence s. The value of n, which is the size of Hankel blocks, parameterizes the extent to which the temporal information is encoded in one feature vector $\mathbf{h}_i \in \mathbb{R}^{f \times n}$, where $\mathbf{h}_i = [\mathbf{x}_i^{\mathrm{T}} \ \mathbf{x}_{i+1}^{\mathrm{T}} \ ... \ \mathbf{x}_{i+n-1}^{\mathrm{T}}]^{\mathrm{T}}$ is the ith column vector of the Hankel matrix $\mathbf{H}$.

The form of the Hankel matrix with block size n corresponds to the RTW matrix formed by the set of TE

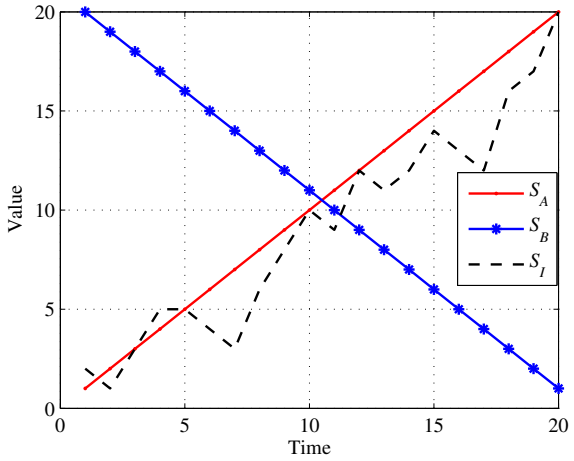

Figure 5: A simple case of univariate reversed sequences and an input sequence with noise. S_A and S_B are two sequences of different dynamics which are exactly the reverse of each other. S_I is an input sequence with similar dynamics to S_A but containing noise.

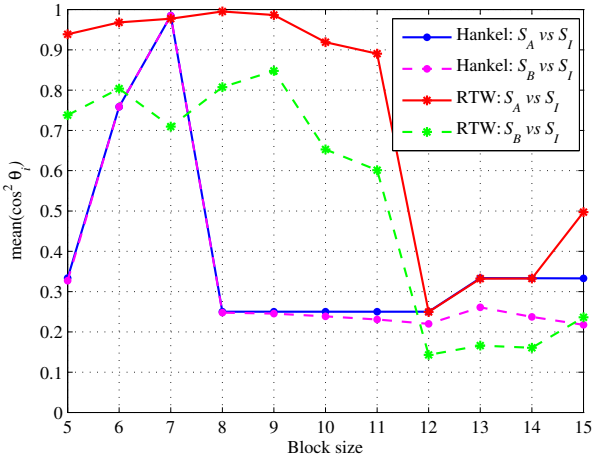

Figure 6: Similarity values between S_A and S_I and between S_B and S_I, using the Hankel method and RTW, with 99% cumulative energy ratio of PCA. With RTW, the similarities between S_A and S_I were higher than those between S_B and S_I. With the Hankel method, when the block size was small, the similarities between S_A and S_I were almost the same as the similarities between S_B and S_I.

features with n images selected. The differences between these two matrices are as follows. Firstly, the maximum number of features in the Hankel method (number of columns) is given by $N^{(s)} - n + 1$, while in RTW it is given by $\binom{N^{(s)}}{n}$. Secondly, the elements of the Hankel matrix are generated by the rule shown in (7), while in RTW the TE features are generated by random sampling. These two differences suggest that the Hankel matrix requires a longer sequence and a much larger number of training sequences than the RTW matrix to generate a rich spatiotemporal dictionary of a motion. Moreover, the Hankel matrix is able to contain only limited global temporal information about a motion, where the extent of globality depends on the size of Hankel blocks.

To demonstrate the advantage of the generalization over the Hankel matrix, we consider the following simple case of toy data. Let $S_A = \{1, \ldots, 20\}$ and $S_B = \{20, \ldots, 1\}$ be two reversed univariate sequential data which belong to two different dynamics. Let S_I be an input sequence similar to S_A but with noises, shown in Figure 5.

A subspace can be used to model the Hankel matrix [10]. The set of the TE features generalizes the Hankel matrix through the bagging-like random sampling. This suggests that the matrix of the set of the TE features contains the dynamical system with some perturbation of the original dynamical system itself. Consequently, subspaces generated from the set of the TE features contain richer information than that from the Hankel matrix and intuitively the subspace representation is also suitable for capturing the information embedded in the set of the TE features. We generated subspaces from the Hankel representation and the set of the TE features of S_A, S_B, and S_I with various block parameters $(5, \ldots, 15)$. The number of the random sampling for RTW was set to 1000. Figure 6 displays the plot of the similarity values when the dimensions of the subspaces were determined by using 99% cumulative energy ratio of PCA. Here, we can see that when using Hankel, the similarity value between S_I and S_A and the similarity value between S_I and S_B were almost the same when the block size was small. Note that this happened because the full rank subspaces that span the trajectory of S_A and S_B are overlapped each other. With the randomization in RTW, S_I becomes more similar to S_A than to S_B.

4. Flow of the recognition framework

Figure 7 shows the flow of the recognition process. In this figure, R indicates the number of TE feature vectors and K_c indicates the number of image sequences of class c.

Training phase:

Step 1 : The random selection is applied to sequence 1 of class c to generate a set of TE

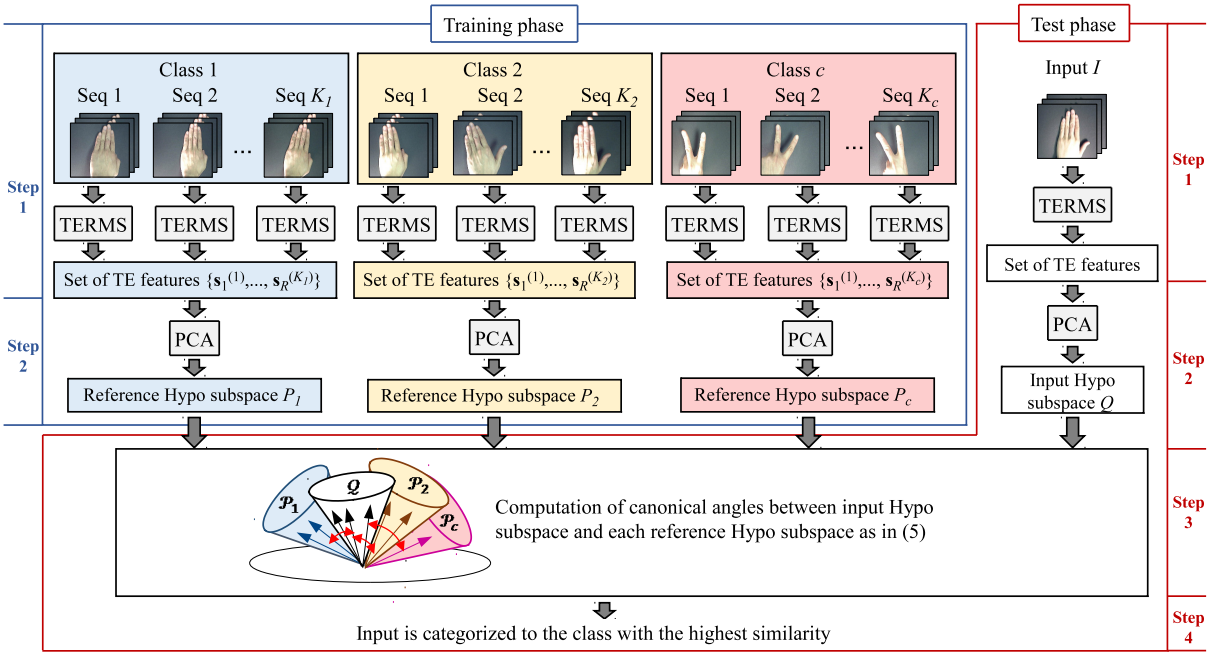

Figure 7: Flow of a recognition process using the RTW framework. TERMS, standing for time elastic random selection, corresponds to the random sampling procedure for generating TE features.

feature vectors $\{\mathbf{s}_1^{(1)}, ..., \mathbf{s}_R^{(1)}\}$. In the case that there is more than one training sequence in one class, this process is done for all K_c sequences to produce a final set of TE feature vectors, $\{\mathbf{s}_1^{(1)}, ..., \mathbf{s}_R^{(1)}, \mathbf{s}_1^{(2)}, ..., \mathbf{s}_R^{(2)}, ..., \mathbf{s}_1^{(K_c)}, ..., \mathbf{s}_R^{(K_c)}\}$.

Step 2 : Reference subspaces $\mathcal{P}_1, ..., \mathcal{P}_c$ of classes $1, ..., c$ are constructed by applying PCA to the corresponding final sets.

Test phase:

Step 1 : Random selection is applied to the sequence of an input motion I to generate a set of TE feature vectors $\{\mathbf{s}_1^{(I)}, \mathbf{s}_2^{(I)}, ..., \mathbf{s}_R^{(I)}\}$.

Step 2 : Input subspace $\mathcal{Q}$ is constructed by applying PCA to the set of TE feature vectors $\{\mathbf{s}_1^{(I)}, \mathbf{s}_2^{(I)}, ..., \mathbf{s}_R^{(I)}\}$.

Step 3 : The similarity between the input subspace $\mathcal{Q}$ and each reference subspace $\mathcal{P}_c$ is computed as in (5).

Step 4 : The input motion I is allocated to the class with the highest similarity:

$$Class(I) = \arg\max_c Sim(\mathcal{Q}, \mathcal{P}_c) . \qquad (8)$$

5. Discriminant analysis of Hypo subspaces

RTW compares the subspaces of the sets of the TE features that were generated through repetition of random sampling, which is an ensemble-like strategy (i.e. a bagging-like sub-sampling but without replacement). The selected features from random sampling can contain irrelevant features that do not contribute to classification. This suggests that the subspaces also contain such features. To suppress the effect of this kind of features, further feature extraction is required. In this section, we briefly describe an adaptation of Grassmann discriminant analysis (GDA) [7] as one of subspace learning methods to improve the discriminative power in the classification task.

A Grassmann manifold $\mathcal{G}(N, d)$ is defined as a set of N-dimensional subspaces of $\mathbb{R}^d$. Hypo subspaces generated from the sets of TE features are considered as points on a Grassmann manifold, where the canonical angles are the distances between them. By repeatedly generating a set of TE features through the random sampling, we can generate multiple reference subspaces that belong to the same class. Then, we apply discriminant analysis that maximizes the variation between classes and minimizes the variation within classes. Let $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^n$ and $\mathbf{Y} = \{y_i\}_{i=1}^n \in \{1, \ldots, C\}$ be a pair set of samples and their class labels, respectively. Discrimi-

nant analysis searches for a transformation matrix $\mathbf{W}$ by maximizing the following function:

$$J(\mathbf{W}) = \frac{\mathbf{W}^\top \mathbf{S}_b \mathbf{W}}{\mathbf{W}^\top \mathbf{S}_w \mathbf{W}}, \tag{9}$$

where $\mathbf{S}_b = \sum_{c=1}^{C} n_c(\mu_c - \mu)(\mu_c - \mu)^\top$ and $\mathbf{S}_w = \sum_{c=1}^{C} \sum_{y_i \in c} (\mathbf{x}_i - \mu_c)(\mathbf{x}_i - \mu_c)^\top$ are the variance of between classes data and the variance of within classes data, respectively; n_c is the number of samples in class c; μ_c and μ are the mean of samples for class c and the mean of all samples, respectively. $\mathbf{W}$ can be obtained by computing the corresponding eigenvectors of the $C-1$ largest eigenvalues of $\mathbf{S}_w^{-1}\mathbf{S}_b$. To apply discriminant analysis on a Grassmann manifold, kernel discriminant analysis [28, 29] with the Grassmann kernel is used [7].

Let $\phi(\mathbf{x})$ be a function that map $\mathbf{x}$ to a Grassmann space $\mathcal{G}$. GDA searches for a mapping $\hat{\mathbf{W}} : \phi(\mathbf{x}) \to \mathbf{y}$ that maximizes the variance of between class data and minimizes the variance of within class data. Since data point $\phi(\mathbf{x})$ on a Grassmann manifold is basically a Hypo subspace which has nontrivial representation, we want to avoid a direct usage of $\phi(\mathbf{x})$. For this purpose, we introduce a kernel function that defines the distance between $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$ as $k(\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)) = \|\phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j)\|_F^2$ which can adopt either (4) or (5). Let the solution be $\hat{\mathbf{W}} = [\hat{\mathbf{w}}_1, \ldots, \hat{\mathbf{w}}_{C-1}] \in \mathbb{R}^{n \times (C-1)}$ and $\hat{\mathbf{w}}_j$ be written as a linear combination of the training data $\hat{\mathbf{w}}_j = \sum_{i=1}^{n} \alpha_{i,j} \phi(\mathbf{x}_i)$. As the result, the mapped ith reference Hypo subspace to the discriminant space is $\mathbf{y}_i = \hat{\mathbf{W}}^\top \phi(\mathbf{x}_i) = \alpha^\top \mathbf{K}_i$, where α is a matrix with size $n \times (C-1)$ and $\mathbf{K}_i$ is the ith column of a kernel matrix $\mathbf{K} \in \mathbb{R}^{n \times n}$ with elements computed from the kernel function $k(\phi(\mathbf{x}_i), \phi(\mathbf{x}_j))$. Equation (9) becomes

$$J(\alpha) = \frac{\alpha^\top \mathbf{K}(V - 1_n 1_n^\top / n)\mathbf{K}\alpha}{\alpha^\top \mathbf{K}(\mathbf{I}_n - V)\mathbf{K}\alpha}, \tag{10}$$

where $1_n \in \mathbb{R}^n$ is a vector with value of 1 in all of its elements, and V is a block diagonal matrix with uniform value of $1_{n_c} 1_{n_c}^\top / n_c$ in the c-th block. The solution of (10) is solved in the same way as for (9).

When adapting GDA to the classification framework, the procedure of the classification is slightly changed as we need to generate multiple reference subspaces from multiple sets of TE features of the same class to be used in the GDA computation. In the classification process, all the reference and input subspaces are mapped to the discriminant space and the classification is then performed by using k-NN based on the Euclidean distance between the mapping results [7].

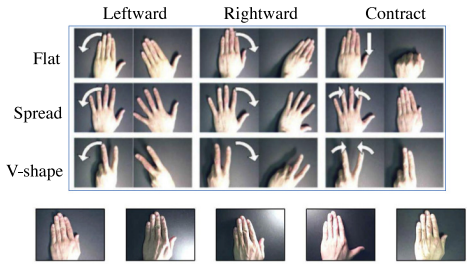

Figure 8: Nine classes and five illumination settings in the Cambridge gesture database [11].

6. Experiments

6.1. Cambridge gesture database

The Cambridge gesture database consists of 9 classes of hand motions which were captured under 5 different illumination settings, as shown in Figure 8. Each class consists of 20 sequences with different numbers of images.

6.1.1. Experimental setup

We conducted an experiment using a setup similar to that in [11, 10], except that in [11] and [10] the length of each image sequence is normalized to a fixed number, while in our case the lengths can be different. The shortest sequence length is 37 and the longest is 119. We resized original images to be of 16×12 pixels and used the grayscale pixel value as the image feature. As a result, the dimension of vector $\mathbf{x}_i$ was 16×12 ($f = 192$). The dimension of the TE feature vector $\mathbf{s}_i$ was $192 \times n$, where n is the number of images obtained by random selection. We used all 20 sequences in the normal illumination setting (Set 5) for training, and the remaining sequences in other illumination settings (Sets 1 to 4) for testing. The total number of test sequences was 720 (9 classes $\times$ 4 sets $\times$ 20 sequences). If not specifically mentioned in the experimental results, the dimension N of reference subspaces was varied from 1 to 60 and the dimension M of an input subspace was varied from 1 to 5. In the experiments using DTW, we first computed the alignment cost between input sequence and the reference sequences. Then, we used k-NN of the alignment costs to decide the class of the input sequence. The results reported here are the best among the parameter settings.

6.1.2. Experimental results

Firstly, we evaluated the effect of the number of canonical angles on the recognition performance. The dimension of the input subspace was fixed at 10. The

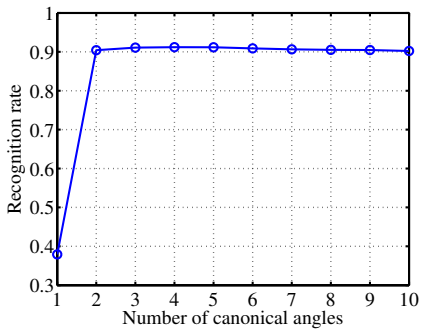

Figure 9: Influence of the number of canonical angles used.

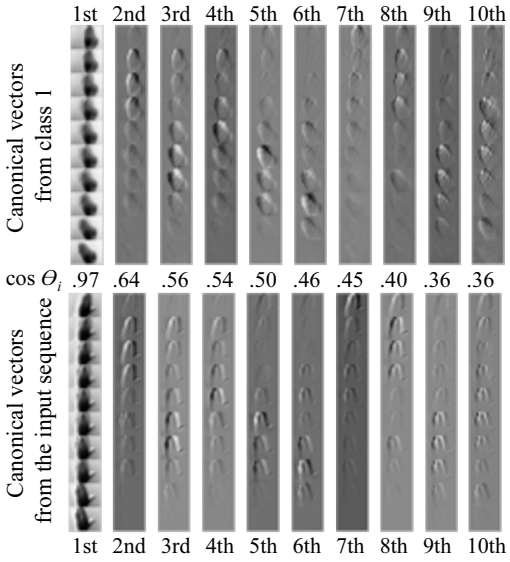

Figure 10: Ten pairs of canonical vectors that form 10 canonical angles between the reference subspace of class 1 and an input subspace of class 1. The average similarity is 0.52.

dimension of the reference subspace was varied from 10 to 60, and the best results are reported here. With this setup, up to 10 canonical angles can be used for the calculation of the similarity measure. The number of selected images for one TE feature, n, was 10. The size of the random selection, R, was 100. From the results shown in Figure 9, we can see that using more than one canonical angle is significantly better than using only the first canonical angle, and the performance achieved by using more than one angle is relatively stable.

To further validate the use of multiple canonical angles, 10 pairs of canonical vectors, which formed 10 canonical angles between an input Hypo subspace of class 1 and the reference Hypo subspaces of classes 1 and 2, are shown in Figures 10 and 11, respectively. In the Cambridge dataset, class 1 is the gesture of a flat

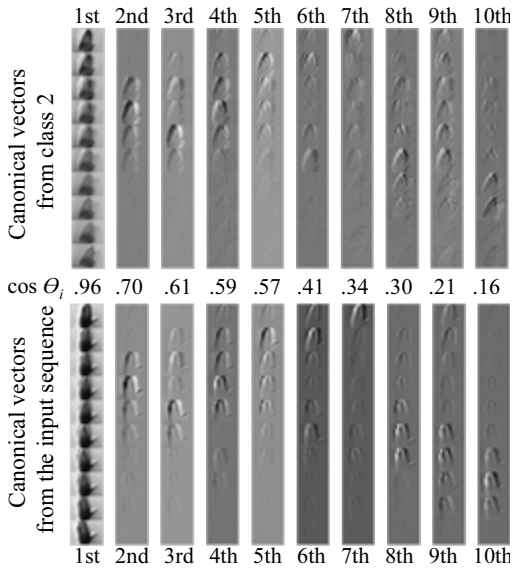

Figure 11: Ten pairs of canonical vectors that form 10 canonical angles between the reference subspace of class 2 and an input subspace of class 1. The average similarity is 0.48.

hand moving leftward, while class 2 is the gesture of a flat hand moving rightward. From both Figures 10 and Figure 11, we can see that the difference between the pairs becomes more noticeable with an increase in the order of the canonical vectors. This suggests that by using multiple canonical vectors we can compare Hypo subspaces more effectively than by using only the first pair of canonical vectors.

Figure 12 shows the value of the similarity (the average of the cosines of the multiple canonical angles) between an input Hypo subspace and each reference Hypo subspace. The difference in the similarity between classes was very small when only the first canonical angle was used: all the similarity values were close to 1. In contrast, by considering multiple canonical angles, the separation of the similarity values between classes increased remarkably. For the rest of our experiments, we used the average of all the canonical angles, which produces approximately the best performance, as shown in Figure 9.

Secondly, we investigated the effect of the number of replicates in the random selection, R, by changing its value to $5, 10, ..., 200$. Figure 13 shows that the recognition rate becomes stable when the value of R reaches about 30.

Thirdly, we compared our proposed RTW method with related methods [11, 8] including DTW-based and the Hankel-based methods. In the case of the Hankel-based method, we used several values of the Hankel

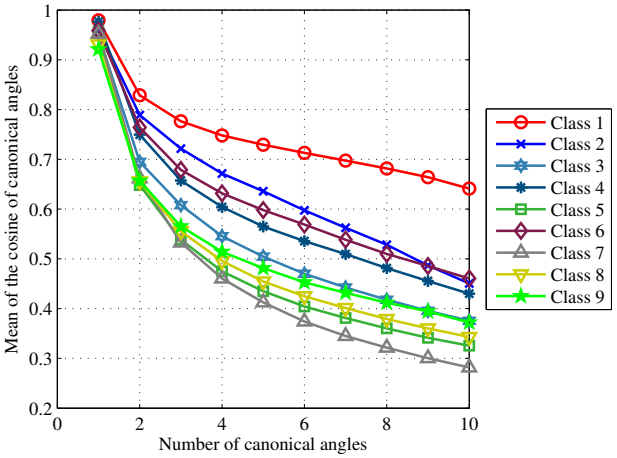

Figure 12: Comparison of similarities between an input subspace of class 1 and each reference class.

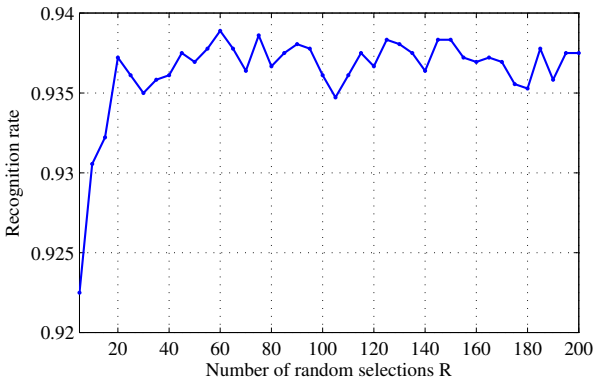

Figure 13: Influence of the number of random selections R on the performance of the proposed method.

block parameter, n, which corresponds to the number of randomly selected images in RTW for one TE feature, ranging from $5, 10, 15, \ldots, 30$. The best results for RTW were obtained with $n = 15$, while the Hankel method produced the best results with $n = 20$. In RTW, the number of random selections R was set to 100. The experiments were repeated 5 times and we computed the average recognition rate. From Table 1, we can confirm that our proposed method outperformed DTW significantly. The performance of DTW was very poor because DTW does not consider variations other than temporal structure. When we performed other experiments for DTW using leave-one-out experimental setting (i.e., 1 sequence from each class was picked for the test, and the rest were used as templates), DTW could achieve recognition rate of 87.3%. Despite the

Table 1: Recognition rates for the Cambridge gesture database [%].

	RTW	DTW	Iso-CCA [20]	Hankel	Kim and Cipolla [11]	Lui and Beveridge [8]
Set 1	95.6	52.8	44.4	94.4	81	93
Set 2	92.9	21.7	13.3	91.1	81	88
Set 3	92.2	26.1	21.1	94.4	78	90
Set 4	93.8	47.2	38.3	91.7	86	91
Avg.	**93.6**	36.9	29.3	92.9	82	91

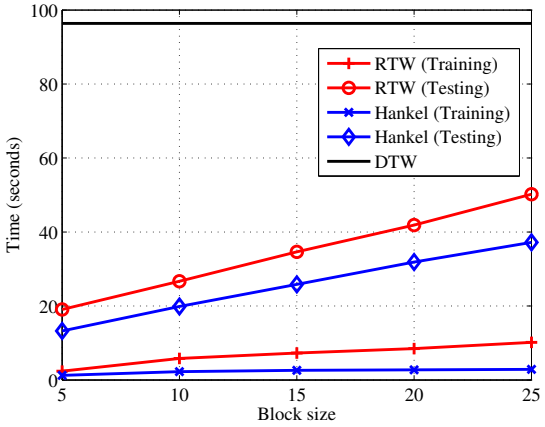

Figure 14: Computational time required to complete the classification experiment on the Cambridge gesture dataset.

easier experimental setting, as the sequences with normal lighting conditions were also used as templates, the recognition rate was still lower than RTW. When we used Isotonic CCA by using the publicly available code from [20], the recognition rate was worse than DTW. We also tested with the publicly available code of CTW [21] and GTW [22] but could not achieve better results than those of [20], where the average recognition rate for CTW and GTW were 21.11% and 27.36%, respectively. In addition, the proposed RTW method is superior to other methods that used tensor representation [11] and tangent spaces [10]. It is also an advantage that our RTW method can handle the unequal lengths of image sequences which commonly exist in motion videos. In contrast, it is difficult to deal with multiple sequences that contain different numbers of images using the conventional methods from [11, 10].

Finally, Figure 14 shows the computational time required to complete the classification of 720 gestures in the Cambridge gesture dataset. All implementations were done using Matlab on Intel Xeon E5-2630 2.3Ghz with 32GB RAM without using parallelization toolbox and the images of the sequential data were already resized into 12×16 pixel. DTW requires no training but it took almost 96 seconds to complete the experiment

(13.39ms per gesture). This is because DTW required to do pairwise comparison between each input gesture with many references. When using the Hankel-based method and RTW with block size of 25, the time required to classify one gesture was about 5.17ms and 6.98ms, respectively. Although there is a slight trade-off in terms of computational time when using RTW, we consider that its recognition phase is still fast enough to be used in a real time application. When conducting experiments with Isotonic CCA, CTW and GTW, the time required to complete the experiments were much more than those of DTW, the Hankel based and RTW, as they require to solve optimization problems that are computationally demanding.

6.2. ChaLearn gesture dataset

The experiments using the Cambridge gesture dataset can be regarded as a case with adequate training samples (20 training samples for each class. In this experiment, we demonstrate the validity of the proposed method for a case with limited training samples (one training sample for each class) by using the dataset from the One-Shot Learning ChaLearn gesture Challenge [5]. The dataset consists of up to 50,000 gestures captured using Microsoft Kinect, grouped into batches. Each batch contains about 100 motions in 8 to 13 gesture categories. The training video contains one gesture, while the test video contains 1 to 5 gestures conducted consecutively. In this experiment we merged 20 batches of the development dataset for which the temporal segmentation and the true labels are provided. Figure 15 shows some examples of motions from the dataset. In the end, after discarding sequences that contain less than 15 frames, the number of classes was 178 and the number of test sequence was 1,557.

6.2.1. Experimental setup

As our focus was on how RTW could improve the performance of conventional methods, we did not use any complicated features. We used the baseline feature extraction of motion histograms of sequential depth images, for which the code was provided by the Challenge [5]. First, a difference depth image is obtained by subtracting two consecutive images. Then, the subtracted depth image is rescaled and vectorized into a 192-dimensional motion histogram feature vector, and the Hypo subspaces are generated by applying PCA to the sets of TE features of the motion histogram feature vectors. The dimensions of subspaces were varied from 1 to 60. However, since the subspace dimension of the Hankel method was limited to $N^{(s)} - n$, where $N^{(s)}$ is the number of frames and n is the size of Hankel blocks, the subspace dimensions for the Hankel method were set to $\min_s(\{N^{(s)}\}) - n$. The number of random selections R was set to 100. Again, the experiments using RTW were repeated 5 times and the average recognition rate is reported as the final result.

In the experiment, we also incorporated a subspace learning method using Grassmann discriminant analysis (GDA) [7] into the framework of RTW, as described in Section 5. As in conventional linear discriminant analysis, GDA needs a lot of training samples. Since RTW generates many TE features, we can provide multiple sets of TE features from one sample sequence which can be used to generate multiple reference Hypo subspaces for the GDA.

6.2.2. Experimental results

Figure 16 summarizes the experimental results from the ChaLearn gesture dataset. Using DTW, we achieved a recognition rate of 63.5%. With Isotonic CCA [20], we obtained a recognition rate of 59.7%. The best result of the Hankel method, with a 72.8% recognition rate, was achieved when using block size 3. We obtained the best recognition rate of 73% when we used the proposed TE features with 6 selected images. The experimental results suggest that, DTW experiences difficulties when there is only one training data item for each class. In the Hankel method, the performance worsens when the size of Hankel blocks increases; this could be due to the fact that in this case the size of Hankel vectors that can be generated from a motion becomes more limited. When we used GDA with 3 subspaces (RTW+GDA3), we obtained the best result of 74.2% when using block size 5. When the number of the subspaces was increased to 5 (RTW+GDA5), the performance was almost the same with that of the RTW+GDA3. From these results we can see that the performance of RTW was improved by incorporating a subspace learning algorithm in the classification framework.

Although our experimental results here is slightly optimistic, since we used the best results obtained from a number of parameter choices, the performance is relatively stable. Figure 17 shows a plot of the recognition rate for various subspace dimensions. The performance stabilized when the subspace dimension reached 30.

We note that the proposed method does not incorporate a function for motion segmentation as a preprocessing, although it is possible. Thus, we could not conduct a direct comparison of the proposed method with other state-of-the-art methods with more complicated functions, such as [30, 31, 32]. The performance metric based on the recognition rate, which is used in

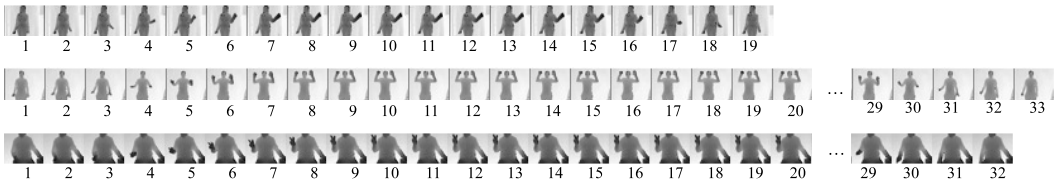

Figure 15: Some examples of the depth image sequences of the ChaLearn gesture dataset from batch 1 class 8 (top), batch 2 class 8 (middle), and batch 3 class 8 (bottom).

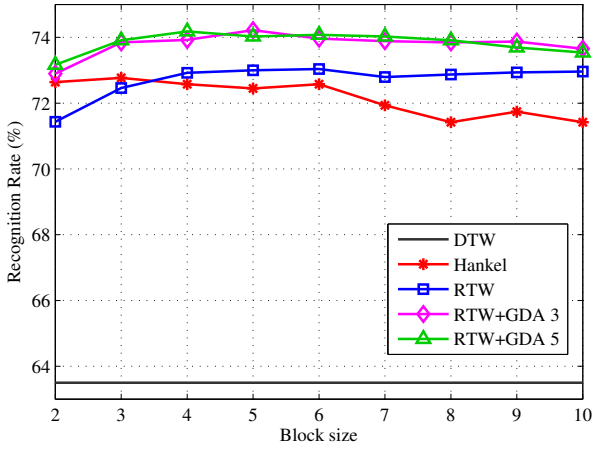

Figure 16: Experimental results for the merged ChaLearn gesture dataset.

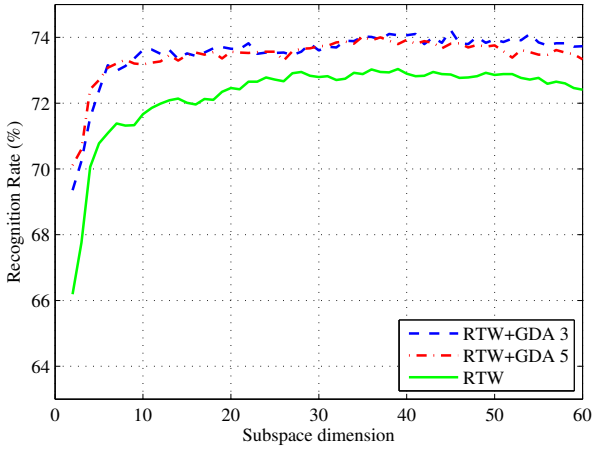

Figure 17: Plot of recognition rate against the dimension of subspace. The blue and red dashed lines are the plots for RTW+GDA3 and GDA5 with TE block size 5, respectively. The green solid line is the plot for RTW with TE block size 6.

our experiments, corresponds to that of the opposite of the edit distance (the Levenshtein distance). This is because we used the ground truth segmentation and consequently the edit distance is the same as the number of miss-classifications (error rate or 1 - recognition rate). We consider the incorporation of a segmentation function into our framework and the comparison with the other more complicated methods as one of the future works.

6.3. KTH action dataset

In this section, we demonstrate the performance of the proposed method using the widely used KTH action dataset [6]. The KTH action dataset [6] consists of six actions: boxing, hand clapping, hand waving, running, jogging, and walking, conducted by 25 subjects under four scenarios: outdoors, outdoors with variation of zooming, outdoors with different clothes, and indoor. In total there are 2, 391 sequences of actions.

6.3.1. Experimental setup

We used the leave-one-out cross validation (LOOCV) scheme, where one sequence was used as the test input and the rest as the training sequences. For each sequence, we used the bounding box from [33] to do segmentation between actions and resize each original frame to a 16×16 pixels grayscale image. We used the raw pixel values with additional information of the height and width of the bounding box of the subject, resulting in a 258-dimensional vector for each frame. We then generated a subspace from each sequence through either the Hankel based method or RTW. The number of the random sampling and the block size were empirically set to 500 and 5, respectively. The dimensionality of the subspaces was set in the same manner as our Chalearn gesture experiment. GDA was adopted as the subspace learning method for both the Hankel and RTW.

6.3.2. Experimental results

Table 2 shows the experimental results for the KTH action dataset. The proposed RTW method achieved recognition rate of 93.39%, outperforming the Hankel method and the established methods in [35, 34]. Our

Table 2: Recognition rates for the KTH action dataset using LOOCV.

Methods	Recognition Rate [%]
Hankel (block size 5)+GDA	91.97
RTW (block size 5)+GDA	93.39
Zhang et al. [34]	90.2
Jiang et al. [33]	93.43
Wang et al. [35]	92.43

proposed method has achieved the same level of performance as [33]. In this comparison, we should note that [33] used a sophisticated shape-motion descriptor that requires multiple processing steps such as silhouette extraction and optical flow computation.

7. Conclusions and future work

In this paper we have proposed RTW, which is a very simple yet effective generalization of the DTW concept, using random sampling, subspaces and multiple canonical angles for classification of sequential data. The matrix of the set of TE features can also be regarded as a generalization of the Hankel matrix. The proposed method addresses the issues of classical DTW, as well as the common issue of lack of training samples that most existing approaches are struggling with. The effectiveness of the proposed method was demonstrated through experiments on the Cambridge gesture database, a subset of the ChaLearn gesture dataset, and the KTH action dataset.

While RTW does not output alignment path, RTW produces multiple similarities and canonical vectors which we regard as the most similar *pseudo*-warped patterns. One task that we have to do is to investigate further the relationship between the canonical vectors and the alignment path in DTW. It is also desirable to know the optimal number of frames or images needed for constituting an effective TE feature vector. We regard this as one of our future works.

We focus on the basic idea of RTW, which combines the random sampling approach with the subspace-based method to act as a generalization of DTW in the conceptual level and the Hankel matrix in the implementation. For this reason, we did not use complicated representation of an image and thus also did not compare the proposed method with the state-of-the-art action recognition methods. Our experimental results can be considered as the baseline performance of RTW. There is room for improving the proposed method. The TE features generated by random selection may contain some information that does not contribute much to recognition. Such information can be better suppressed by applying or developing more sophisticated sampling schemes and feature-extraction techniques. The former includes developing content-based adaptive sampling, which we will consider as a future work; the latter includes nonlinear subspace-based and discriminative learning methods [7, 26], as demonstrated in our experiments on the ChaLearn gesture dataset and the KTH action dataset. Moreover, adopting sophisticated feature extraction for each frame or a group of frames such as [33] prior to the random sampling is also one direction which may further improve the capability of the proposed method, especially for categorizing actions in challenging situations.

Acknowledgment

We would like to thank Dr. Yasuhiro Ohkawa, Dr. Hideitsu Hino, and Prof. Seiichi Uchida for discussions and suggestions. This work was supported by JSPS KAKENHI Grant (No. 25282173).

References

[1] S. B. Needleman, C. D. Wunsch, A general method applicable to the search for similarities in the amino acid sequence of two proteins, Journal of Molecular Biology 48 (1970) 443–53.

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

[3] T. Darrell, A. Pentland, Space-time gestures, in: IEEE Conference on Computer Vision and Pattern Recognition, 1993, pp. 335–340.

[4] T. Kim, R. Cipolla, Canonical correlation analysis of video volume tensors for action categorization and detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (2009) 1415–1428.

[5] ChaLearn gesture dataset (CGD2011), 2011. ChaLearn, California.

[6] C. Schuldt, I. Laptev, B. Caputo, Recognizing human actions: A local SVM approach, in: International Conference on Pattern Recognition, 2004, pp. 32–36.

[7] J. Hamm, D. D. Lee, Grassmann discriminant analysis: a unifying view on subspace-based learning, in: International Conference on Machine Learning, 2008, pp. 376–383.

[8] Y. Lui, J. Beveridge, Action classification on product manifolds, in: IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 833–839.

[9] M. T. Harandi, C. Sanderson, S. Shirazi, B. C. Lovell, Kernel analysis on Grassmann manifolds for action recognition, Pattern Recognition Letters (2013).

[10] B. Li, M. Ayazoglu, T. Mao, O. Camps, M. Sznaier, Activity recognition using dynamic subspace angles, in: IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 3193–3200.

[11] T. Kim, J. Kittler, R. Cipolla, Discriminative learning and recognition of image set classes using canonical correlations, IEEE Transaction on Pattern Analysis and Machine Intelligence 29 (2007) 1005–1018.

[12] S. Nakagawa, H. Nakanishi, Speaker-independent English consonant and Japanese word recognition by a stochastic dynamic time warping method, Journal of the Institution of Electronics and Telecommunication Engineers 34 (1988) 87–95.

[13] J. F. Lichtenauer, E. A. Hendriks, M. J. T. Reinders, Sign language recognition by combining statistical DTW and independent classification, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (2008) 2040–2046.

[14] C. Bahlmann, H. Burkhardt, The writer independent online handwriting recognition system frog on hand and cluster generative statistical dynamic time warping, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (2004) 1–12.

[15] T. Katayama, G. Picci, Realization of stochastic systems with exogenous inputs and subspace identification methods, Automatica 35 (1999) 1635–1652.

[16] Y. Kawahara, T. Yairi, K. Machida, A kernel subspace method by stochastic realization for learning nonlinear dynamical systems., in: Neural Information Processing Systems, 2006, pp. 665–672.

[17] J. Aggarwal, M. Ryoo, Human activity analysis: a review, ACM Computing Survey 43 (2011) 16:1–16:43.

[18] R. Poppe, A survey on vision-based human action recognition, Image and Vision Computing 28 (2010) 976–990.

[19] P. F. Felzenszwalb, R. Zabih, Dynamic programming and graph algorithms in computer vision., IEEE Transaction on Pattern Analysis and Machine Intelligence 33 (2011) 721–740.

[20] S. Shariat, V. Pavolovic, Isotonic CCA for sequence alignment and activity recognition, in: International Conference on Computer Vision, 2011, pp. 2572–2578.

[21] F. Zhou, F. De la Torre Frade, Canonical time warping for alignment of human behavior, in: Neural Information Processing Systems, 2009, pp. 1–9.

[22] F. Zhou, F. De la Torre Frade, Generalized time warping for multi-modal alignment of human motion, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1282–1289.

[23] K. Schindler, L. Van Gool, Action snippets: how many frames does human action recognition require?, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.

[24] K. Maeda, S. Watanabe, Pattern matching method with local structure, IEICE Transaction J68-D (1985) 345–352.

[25] K. Fukui, O. Yamaguchi, Face recognition using multi-viewpoint patterns for robot vision, in: International Symposium on Robotics, 2003, pp. 192–201.

[26] K. Fukui, A. Maki, Difference subspace and its generalization for subspace-based methods, IEEE Transactions on Pattern Analysis and Machine Intelligence 37 (2015) 2164–2177.

[27] Y. Ohkawa, K. Fukui, Hand-shape recognition using the distributions of multi-viewpoint image sets, IEICE Transaction E95-D (2012) 1619–1627.

[28] S. Mika, G. Ra, J. Weston, B. Scho, A. Smola, Constructing descriptive and discriminative nonlinear features : Rayleigh coefficients in kernel feature spaces, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (2003) 623–628.

[29] D. Cai, X. He, J. Han, Speed up kernel discriminant analysis, The VLDB Journal 20 (2011) 21–33.

[30] Y. M. Lui, Human gesture recognition on product manifolds, J. Mach. Learn. Res. 13 (2012) 3297–3321.

[31] S. R. Fanello, I. Gori, G. Metta, F. Odone, Keep it simple and sparse: Real-time action recognition, J. Mach. Learn. Res. 14 (2013) 2617–2640.

[32] N. A. Goussies, S. Ubalde, M. Mejail, Transfer learning decision forests for gesture recognition, J. Mach. Learn. Res. 15 (2014) 3667–3690.

[33] Z. Jiang, Z. Lin, L. Davis, Recognizing human actions by learning and matching shape-motion prototype trees, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (2012) 533–547.

[34] X. Zhang, Y. Yang, L. C. Jiao, F. Dong, Manifold-constrained coding and sparse representation for human action recognition, Pattern Recognition 46 (2013) 1819–1831.

[35] Y. Wang, P. Sabzmeydani, G. Mori, Semi-latent Dirichlet allocation: A hierarchical model for human action recognition, in: Procc. ICCV Workshop on Human Motion Understanding, Modelling, Capture and Animation, 2007, pp. 240–254.