Enhanced Grassmann Discriminant Analysis with Randomized Time Warping for Motion Recognition

Lincon Souza[a,*], Bernardo B. Gatto[b], Jing-Hao Xue[c], Kazuhiro Fukui[a]

[a]*Graduate School of Systems and Information Engineering, University of Tsukuba, Tsukuba-shi, Japan.*
[b]*Center for Artificial Intelligence Research (C-AIR), University of Tsukuba, Tsukuba-shi, Japan.*
[c]*Department of Statistical Science, University College London, London, United Kingdom.*

Abstract

This paper proposes a framework for classifying motion sequences, by extending the framework of Grassmann discriminant analysis (GDA). A problem of GDA is that its discriminant space is not necessarily optimal. This limitation becomes even more prominent when utilizing the subspace representation of randomized time warping (RTW). RTW is a sequence representation that can effectively model a motion's temporal information by a low-dimensional subspace, simplifying the problem of comparing two sequences to that of comparing two subspaces. The key idea of the proposed enhanced GDA is projecting class subspaces onto a generalized difference subspace before mapping them on a Grassmann manifold. The GDS projection can remove overlapping components of the subspaces in the vector space, nearly orthogonalizing them. Consequently, a dictionary of orthogonalized class subspaces produces a set of more discriminant data points in the Grassmann manifold, in comparison with the original set. This set of data points can further enhance the discriminant ability of GDA. We demonstrate the validity of the proposed framework, RTW+eGDA, through experiments on motion recognition using the publicly available Cambridge gesture, KTH action, and UCF sports datasets.

Keywords: enhanced GDA, randomized time warping, motion recognition
2010 MSC: 00-01, 99-00

1. Introduction

This paper proposes a method for characterizing and classifying motion image sequences, focusing on hand gestures and human actions. We extend the framework of Grassmann discriminant analysis (GDA) [1] to work more effectively in the application of motion recognition. The problem of GDA that we address in this paper is that GDA's discriminant space is not necessarily optimal. This limitation becomes even

*Corresponding author
Email addresses: lincons@cvlab.cs.tsukuba.ac.jp (Lincon Souza), bernard.gatto@gmail.com (Bernardo B. Gatto), jinghao.xue@ucl.ac.uk (Jing-Hao Xue), kfukui@cs.tsukuba.ac.jp (Kazuhiro Fukui)

more prominent when representing motion sequences by the randomized time warping (RTW) [2] subspace representation.

Randomized time warping (RTW) is an effective generalization of dynamic time warping (DTW) [3], which is one of the most widely used methods for motion analysis. The core idea of DTW is to compare two sequences by searching for the best alignment of their sequential patterns; this is performed by optimizing a warping function with dynamic programming. In contrast to DTW, RTW has a compact representation and does not need dynamic programming, thus providing a fast and light algorithm. It converts the problem of comparing two sequences to comparing two low-dimensional subspaces, called sequence hypothesis (hypo) subspaces. This problem can in turn be solved by measuring the canonical angles between them. The mutual subspace method (MSM) [4] is well known as a fundamental classification method using canonical angles, which has been used along with RTW.

Comparison of hypo subspaces has also been performed by introducing the Grassmann manifold formulation, which simplifies the complicated procedure of the mutual subspace method using canonical angles. The Grassmann manifold, symbolized as $\mathcal{G}(m, D)$, is defined as a set of m-dimensional linear subspaces of $\mathbb{R}^D$ [5]. In this framework, a subspace-based method is regarded as a simple classification method on a Grassmann manifold, where each single subspace is treated as a point, and thereby, each motion video is represented by a point in the manifold. Various types of classification methods have been constructed on a Grassmann manifold, such as Grassmann discriminant analysis (GDA) [1], Bayesian classifier on the Grassmann manifold [6], or learning on the manifold [7] . Among them, in particular, RTW formulation has been used along with GDA [2], which has been known as one of the useful tools for image set classification [8, 9]. GDA can be easily conducted as a kernel discriminant analysis through the kernel trick with a Grassmann kernel [10, 11].

Although it has been useful to combine RTW with GDA, some issues arise from this representation:

- same-class actions may have vary large variations, while semantically different actions may have similar movements, making different action subspaces closer to each other, causing overlap among them in the worst case;

- although GDA is capable of finding the most discriminant directions in a manifold with respect to the given data points on the manifold, it cannot operate the corresponding original subspaces in the vector space. Hence, if subspaces were not well separated in vector space, the corresponding data points on the induced manifold are also not adequately separated, in such a way that GDA may not be able to separate them.

To address those problems, the key idea proposed in this paper is to project hypo subspaces onto a generalized difference subspace (GDS) [12], before mapping each class subspace on a Grassmann manifold. GDS is a general concept that represents difference among multiple class subspaces, which forms a discriminative space. GDS projection works effectively as a powerful feature extraction for subspace-based methods such as subspace method [13] and mutual subspace method (MSM) [4], as it can enlarge the angles among class subspaces toward the orthogonal status. These subspace

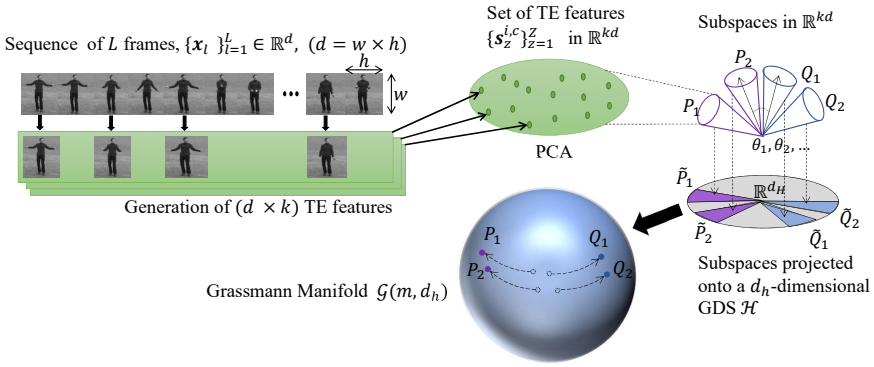


Figure 1: Conceptual diagram of the proposed method. A set of TE features is extracted by randomly sampling images from an image sequence. Next, a hypo subspace is generated by applying PCA to the set. For each image sequence, a hypo subspace is generated in this way. Finally, the hypo subspaces are orthogonalized by projecting them onto the GDS, and then are projected onto the Grassmann Manifold.

methods conduct the classification by using the canonical angles between an input vector/subspace and each reference class subspace. A MSM with GDS projection is called constrained MSM (CMSM) [14]. GDS has also been extended recently, such as by adding regularization [15] and for different applications, i.e. high-dimensional spectral data [16]. The idea of using GDS for motion recognition has been motivated by previous preliminary work in [17, 18] and this paper contains more in-depth analysis with extensive and comprehensive experiments.

It is worth mentioning that other methods have extended the MSM formulation to produce discriminative features. A remarkable example is the discriminative canonical correlation (DCC) [19]. DCC has been motivated in that the structural similarity between class subspaces is measured by the canonical angles between them. Different from CMSM, DCC iteratively computes a discriminative subspace using the Fisher discriminant analysis (FDA) as an objective function to further improves its class separability. Although its exceptional results, DCC's computational time is usually costly. GDS, on the other hand, requires only an SVD computation, which is very efficient in modern implementations.

Figure 1 shows the conceptual diagram of the proposed method. A set of TE features is extracted by randomly sampling images from an image sequence. Next, a hypo subspace is generated by applying PCA to the set. For each image sequence, a hypo subspace is generated in this way. Finally, the relationship among hypo subspaces comes close to the orthogonal status by projecting them onto the GDS, and then the projected subspaces are mapped onto the Grassmann Manifold. The reason for performing GDS projections before mapping each subspace with the Grassmann kernel is that GDS can operate the hypo subspaces directly in the vector space. Concretely, when some data overlaps among multiple classes, GDA's vector representation cannot necessarily distinguish these data, even if they are projected onto the optimal discriminant space found by GDA. In contrast, GDS can remove overlapping components of the subspaces in

the vector space, nearly orthogonalizing them, and as a result creating more discriminant data points for GDA.

As GDS has the function of removing common features among class subspaces, providing more discriminative sample for GDA, it is expected that GDS projection can solve the overlap problem and further enhance the representation of the RTW hypo subspaces on the Grassmann manifold. The validity of our proposed method is demonstrated through experiments with the Cambridge gesture [20], KTH action [21] and UCF sports [22, 23] datasets.

In summary, the main contribution of our method is to provide a simple and practical means for further enhancing the performance of GDA, which has been widely used in various applications. In particular, we introduce GDS projection to the GDA formulation to enhance RTW+GDA by alleviating the problems regarding TE feature generation for RTW.

The rest of the paper is organized as follows. In Sec. 2, we elaborate on the basic idea that leads to our proposed method for classifying motion, explained in detail in Sec. 3. In Sec. 4, we conduct experiments on motion recognition using three public datasets. Sec. 5 concludes the paper.

2. Basic Idea for Enhancing GDA

Our key idea for enhancing Grassmann discriminant analysis (GDA) with hypo subspaces is to project hypo subspaces onto a generalized difference subspace (GDS) before applying GDA to them. In the following, we describe more deeply the problem mentioned in Sec.1 and the mechanism which induces the effective function to address it.

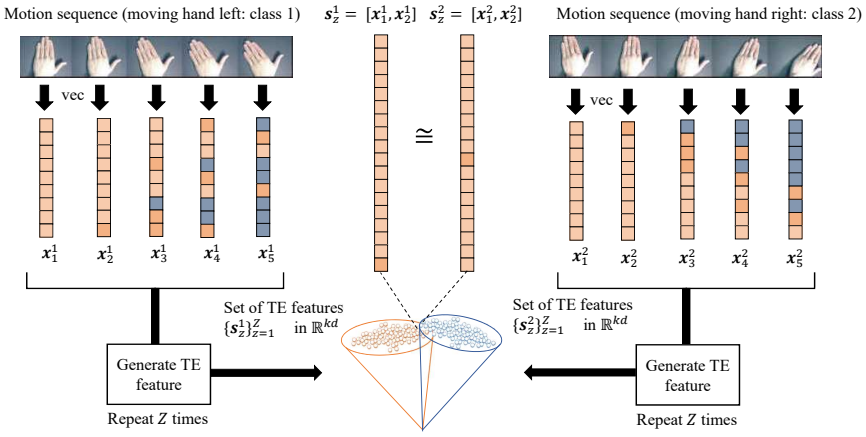


Figure 2: Conceptual diagram of the detailed intuition behind RTW. Two sequences of images of different hand gesture classes (e.g. moving hand left and right) are processed through RTW: the frames are vectorized, and a set of TE features is generated by randomly sampling frames and concatenating. In the middle, two examples of TE feature are examined, one vector from each set, which are very close to each other.

First, we discuss the intuition of RTW and the advantage of utilizing hypo subspaces as a representation for sequences. The core idea of RTW is to generate a set of time warped patterns, called time elastic (TE) features, through repeated random subsampling, while preserving the original temporal order. This mechanism can be regarded as a simultaneous search for the most similar warped patterns from a number of randomly obtained candidates, in contrast to the inefficient search performed by DTW. As the cost of comparing two sets of TE features increases dramatically as the number of features increase, the comparison is conducted using a subspace based method, in which each set of TE features is represented as a hypo subspace. In summary, a hypo subspace is a compact representation for sequences as it is independent of sequence length, while the canonical angles offer a simple tool to calculate the most similar warped patterns between two sequences.

In the following we elaborate on the problem of overlapping hypo subspaces. The reason for proximity in the TE feature space is that some frames lead to similar variables. In practice this may have various reasons: 1) some sampled frames may be motionless and composed of similar texture; 2) or their movement is similar in direction; 3) or important moving parts are occluded. Figure 2 shows a conceptual diagram of the detailed intuition behind RTW. Two sequences of images of different hand gesture classes (e.g. moving hand left and right) are processed through RTW: each frame $\boldsymbol{x}_l^c$ is vectorized, where l denotes frame number and c the class in this example, either 1 or 2. In this simple example, a set of TE features is generated by repeating Z times the process of sampling 2 frames from 5 frames and concatenating the 2 frames. A TE feature is denoted as $\boldsymbol{s}_z^c$, where $z = 1, \ldots, Z$. The center of Fig. 2 shows the case 1), where two examples of TE features are very close to each other. As a result, when the hypo subspaces of the two sets are generated by applying PCA to each set, they are more likely to overlap.

In applications with real unconstrained data, the probability that concatenated frames present a significative amount of correlation becomes high given various conditions, such as: slow motion speed, specially when the action contains moments of idleness or interruptions; and small appearance changes, specially when the moving target object is far from camera, or some parts are occluded. In many cases, more than one of these factors cause TE features to be close to each other.

To solve this problem, one could think a naive approach of calculating the similarities between frames and then removing similar frames between sets; however, the random sampling of RTW is by itself a statistical technique to avoid the need to compare individual frames, as this is not a scalable operation in terms of complexity. In this sense, a desirable solution needs to consider a subspace representation, rather than analyzing the individual TE features or their frames.

Now, we explain the definition and mechanism of GDS and how GDS projection can be harnessed for solving the aforementioned problem. GDS is defined as a subspace, which represents a "difference" among multiple class subspaces [12]. GDS is a further extension of difference subspace (DS) for two class subspaces, which is a natural generalization of a difference vector of two vectors.

Given $C(\geq 2)$ m-dimensional class subspaces, $\{\mathcal{P}_c\}_{c=1}^{C}$, a generalized difference subspace (GDS), $\mathcal{H}$, can be defined as the subspace produced by removing the principal component subspace (PCS) of all the class subspaces from the sum subspace, $\mathcal{S}$, of those

subspaces. This definition of GDS leads GDS projection to the function of automatically removing overlap among class subspaces, which can alleviate the problem. It is worth noting here that we consider removing the overlapping components of the data, not the data themselves. The details of the process of GDS projection will be explained in Sec. 3.3. On the other hand, we should note that GDA cannot necessarily distinguish data belonging to overlap region, even by projecting them onto its optimal discriminant space.

Besides, GDS projection has the function of orthogonalizing class subspaces by enlarging the canonical angles among class subspaces. Although GDA also has a similar function, the mechanisms of both are quite different. GDA works on a Grassmann manifold, while GDS projection works in the original high dimensional vector space before being mapped onto the Grassmann manifold. Based on this difference, we expect different effects from GDS and GDA to learn a discriminant space where the classes are as separated as possible.

In terms of computational complexity, GDS has an advantage over GDA as its complexity is linearly proportional to the number of training subspaces N and cubic with respect to the dimension of the subspaces m. In contrast, the complexity of GDA is quadratically proportional to the number of training subspaces N and cubic with respect to the subspace dimension m. Therefore, the proposed eGDA's complexity effectively maintains the same order of complexity as GDA. Table 1 shows the complexity of each method.

Table 1: Computational complexity of GDA, GDA and the proposed eGDA.

Method	Complexity
GDS	$(N + C + 1)\mathcal{O}(m^3)$
GDA	$(N + \frac{N(N+1)}{2} + 1)\mathcal{O}(m^3)$
eGDA	$(N + \frac{N(N+1)}{2} + C + 2)\mathcal{O}(m^3)$

3. Algorithm of the Proposed Method

We first describe the representation by RTW to generate a hypo subspace; then we explain how to generate a GDS and use its projection to enhance GDA. The step-by-step training and testing algorithms of the proposed method are shown in Algorithms 1 and 2, respectively.

3.1. Motion Sequence Representation by RTW

In our method, an image with the size $w \times h$ is represented by a $d(= w \times h)$-dimensional vector $\boldsymbol{x} \in \mathbb{R}^d$. Consider N_c training ordered sequences $\{\boldsymbol{x}_l^{i,c}\}_{l=1}^{L_i^c}$ for each c-th class ($c = 1, \ldots, C$), where $i = 1, \ldots N_c$ indicate the indices of sequences of the c-th class, and $l = 1, \ldots, L_i^c$ are the indices of individual images of a sequence. Consider also an ordered sequence of L_{in} input images $\{\boldsymbol{x}_l^{in}\}_{l=1}^{L_{in}}$. For example, each of these sequences represent a body motion or hand gesture captured by video.

Algorithm 1: Learning algorithm of the proposed method

input training ordered sequences $\{\boldsymbol{x}_l^{i,c}\}_{l=1}^{L_i^c}$, with class label c
:
for $c = 1, \ldots, C$ **do**
  for $i = 1, \ldots, N_c$ **do**
    $\{\boldsymbol{s}_z^{i,c}\}_{z=1}^{Z} \leftarrow \mathrm{TE}(\{\boldsymbol{x}_l^{i,c}\}_{l=1}^{L_i^c})$ // 1: obtain TE features
    $\boldsymbol{R}_i^c \leftarrow \frac{1}{Z}\sum_{z=1}^{Z} \boldsymbol{s}_z^{i,c}\boldsymbol{s}_z^{i,c^\top}$ // 2: calculate set covariance matrix
    $\boldsymbol{Y}_i^c \leftarrow \mathrm{EVD}(\boldsymbol{R}_i^c)$ // 3: apply eigendecomposition
  end
  $\boldsymbol{R}^c \leftarrow \frac{1}{N_c}\sum_{i=1}^{N_c} \boldsymbol{R}_i^c$ // 4: calculate class covariance matrix
  $\boldsymbol{M}_c \leftarrow \mathrm{EVD}(\boldsymbol{R}^c)$ // 5: apply eigendecomposition
end
$\boldsymbol{P}, \boldsymbol{H} \leftarrow \mathrm{EVD}(\sum_{c=1}^{C} \boldsymbol{M}_c \boldsymbol{M}_c^\top)$ // 6: obtain GDS and principal subspace
foreach $\boldsymbol{Y}_i^c$ **do** $\widetilde{\boldsymbol{Y}}_i^c \leftarrow H^\top \boldsymbol{Y}_i^c$ // 7: project all subspaces onto the GDS
for $q = 1, \ldots, N$ **do**
  for $w = 1, \ldots, N$ **do**
    $[\boldsymbol{S}_{train}]_q^w \leftarrow k_p(\widetilde{\boldsymbol{Y}}_q, \widetilde{\boldsymbol{Y}}_w)$ // 8: generate similarity matrix
  end
end
$\boldsymbol{\alpha}^* \leftarrow \max_{\boldsymbol{\alpha}} Ra(\boldsymbol{\alpha})$ // 9: solve LDA problem
$\boldsymbol{F}_{train} \leftarrow \boldsymbol{\alpha}^{*^\top} \boldsymbol{S}_{train}$ // 10: compute training coefficients
return $\boldsymbol{F}_{train}, \boldsymbol{H}, \boldsymbol{\alpha}^*$ // return dictionary, GDS and GDA projection operators

An $d \times k$ dimensional TE feature vector $\boldsymbol{s} = [\boldsymbol{y}_1^\top \boldsymbol{y}_2^\top \ldots \boldsymbol{y}_k^\top]^\top$ is created by randomly selecting k images from a sequence $\{\boldsymbol{x}_l^{i,c}\}_{l=1}^{L_i^c}$, such that $\boldsymbol{y}_1^\top \boldsymbol{y}_2^\top \ldots \boldsymbol{y}_k^\top \in \{\boldsymbol{x}_l^{i,c}\}_{l=1}^{L_i^c}, t(\boldsymbol{y}_1) < \ldots < t(\boldsymbol{y}_k)$, where $t(\cdot)$ denotes the original order of the image.

Let this procedure of random selection be repeated Z times, such that we obtain $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_Z$. Subsequently, an auto-correlation matrix $\boldsymbol{R}_i^c$, which corresponds to the set of the TE feature vectors, can be computed as:

$$\boldsymbol{R}_i^c = \frac{1}{Z}\sum_{z=1}^{Z} \boldsymbol{s}_z^{i,c}\boldsymbol{s}_z^{i,c^\top}. \tag{1}$$

This procedure corresponds to steps 1 and 2 in Algorithms 1 and 2.

3.2. Subspace Representation

We utilize the principal component analysis (PCA) by computing the eigenvectors of each matrix $\boldsymbol{R}_i^c$ to construct m-dimensional subspaces $\mathcal{Y}_i^c$. The orthonormal basis

Algorithm 2: Input evaluation algorithm of the proposed method

input pattern set with L' input images $\{\boldsymbol{x}^{in}\}$
:
$\{\boldsymbol{s}_z^{in}\}_{z=1}^{Z} \leftarrow \mathrm{TE}(\{\boldsymbol{x}^{in}\})$ // 1: obtain TE features
$\boldsymbol{R}_{in} \leftarrow \frac{1}{Z}\sum_{z=1}^{Z} \boldsymbol{s}_z^{in}\boldsymbol{s}_z^{in\top}$ // 2: calculate set covariance matrix
$\boldsymbol{X} \leftarrow \mathrm{EVD}(\boldsymbol{R}_{in})$ // 3: apply eigendecomposition
$\widetilde{\boldsymbol{X}} \leftarrow \boldsymbol{H}^{\top}\boldsymbol{X}$ // 4: project subspace onto the GDS
for $q = 1, \ldots, N$ **do**
  $[\boldsymbol{S}_{test}]_q \leftarrow k_p(\widetilde{\boldsymbol{Y}}_q, \widetilde{\boldsymbol{X}})$ // 5: generate similarity matrix
end
$\boldsymbol{F}_{test} \leftarrow \boldsymbol{\alpha}^{*\top}\boldsymbol{S}_{test}$ // 6: compute test coefficients
$\mathrm{pred}(\boldsymbol{x}^{in}) \leftarrow \mathrm{NN}(\boldsymbol{F}_{train}, \boldsymbol{F}_{test})$ // 7: perform 1-NN classification
return $\mathrm{pred}(\boldsymbol{x}^{in})$ // return a class prediction

of each subspace are obtained as the eigenvectors corresponding to the m largest eigenvalues. In the following, each m-dimensional subspace $\mathcal{Y}_i^c$ is represented by the matrix $\boldsymbol{Y}_i^c \in \mathbb{R}^{kd\times m}$, which has the corresponding orthonormal basis as its column vectors. A set of TE features generated from a sequence contains various possible warped patterns in time, 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. In Algorithms 1 and 2, the generation of hypo subspaces corresponds to step 3.

3.3. Projection onto Generalized Difference Subspace

In order to utilize the feature extraction function of GDS effectively, we introduce the global class subspaces $\mathcal{M}_c$, which is denoted by a matrix $\boldsymbol{M}_c \in \mathbb{R}^{kd\times d_m}$, which represents compactly all the subspaces belonging to the same class c. The orthogonal basis of $\mathcal{M}_c$ can be obtained as the eigenvectors corresponding to the d_m largest eigenvalues of the auto-correlation matrix:

$$\boldsymbol{R}^c = \frac{1}{N_c}\sum_{i=1}^{N_c}\boldsymbol{R}_i^c = \frac{1}{ZN_c}\sum_{i=1}^{N_c}\sum_{z=1}^{Z}\boldsymbol{s}_z^{i,c}\boldsymbol{s}_z^{i,c\top}. \tag{2}$$

Next, to generate a GDS, we calculate the total sum matrix, $\boldsymbol{S}$, which is defined as:

$$\boldsymbol{S} = \sum_{c=1}^{C}\sum_{j=1}^{d_m}\boldsymbol{\Phi}_j^c\boldsymbol{\Phi}_j^{c\top}, \tag{3}$$

where $\boldsymbol{\Phi}_j^c$ is a basis of the d_m-dimensional $\mathcal{M}_c$. The orthogonal basis of the GDS can be obtained as d_h eigenvectors, $\{\boldsymbol{d}_i\}_{i=1}^{d_h}$ corresponding to the d_h smallest eigenvalues of the sum matrix $\boldsymbol{S}$. The subspaces $\boldsymbol{Y}_i^c$ are projected onto the GDS and their projections are denoted by $\{\widetilde{\boldsymbol{Y}}_i^c\}_{i=1}^{N_c} \in \mathbb{R}^{d_h\times m}$. The input subspace of $\boldsymbol{X}$ is also projected onto the GDS and its projection is denoted by $\widetilde{\boldsymbol{X}}$. In Algorithm 1 the generation of class subspaces and the GDS corresponds to steps 4 to 6, while projection of subspaces is step 7. In Algorithm 2, only projection is performed, corresponding to step 4.

3.4. Enhancing Grassmann Discriminant Analysis

Now, we outline the algorithm of GDA and how we utilize GDS projection to enhance its discrimination capability.

The essence of GDA lies in the concept of the Grassmann manifold $\mathcal{G}(m, d)$, defined as the set of m-dimensional linear subspaces of $\mathbb{R}^d$. It is an $m(d-m)$-dimensional compact Riemannian manifold and can be derived as a quotient space of orthogonal groups $\mathcal{G}(m, d) = \mathcal{O}(d)/\mathcal{O}(m)\times\mathcal{O}(d-m)$, where $\mathcal{O}(m)$ is the group of $m\times m$ orthonormal matrices. The Grassmann manifold can be embedded in a reproducing kernel Hilbert space by the use of a Grassmann kernel. In this case, the most popular kernel is the projection kernel k_p, which can be defined as $k_p(\mathcal{Y}_1, \mathcal{Y}_2) = \sum_{i=1}^{m} \cos^2 \theta_i$. We can measure the distance between two points on a Grassmann manifold by using this projection kernel [10].

Basically, GDA is conducted as kernel LDA with the Grassmann kernels. We first outline the algorithm of linear discriminant analysis (LDA) [24]. Let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ be the data vectors and $y_1, \ldots, y_N$ ($y_i \in 1, \ldots, C$) be the class labels. Each class c has N_c number of samples. Let $\boldsymbol{\mu}_c = \frac{1}{N_c} \sum_{i|y_i=c} \boldsymbol{x}_i$ be the mean of class c, and $\boldsymbol{\mu} = \frac{1}{N} \sum_i \boldsymbol{x}_i$ be the overall mean. LDA searches for the discriminant direction $\boldsymbol{w}$ which maximizes the Rayleigh quotient $Ra(\boldsymbol{w}) = \boldsymbol{w}' S_b \boldsymbol{w} / \boldsymbol{w}' S_w \boldsymbol{w}$ where $\boldsymbol{S}_b$ and $\boldsymbol{S}_w$ are the between-class and within-class covariance matrices respectively:

$$\boldsymbol{S}_b = \frac{1}{N} \sum_{c=1}^{C} N_c (\boldsymbol{\mu}_c - \boldsymbol{\mu})(\boldsymbol{\mu}_c - \boldsymbol{\mu})^\top, \tag{4}$$

$$\boldsymbol{S}_w = \frac{1}{N} \sum_{c=1}^{C} \sum_{i|y_i=c} (\boldsymbol{x}_i - \boldsymbol{\mu}_c)(\boldsymbol{x}_i - \boldsymbol{\mu}_c)^\top. \tag{5}$$

The optimal $\boldsymbol{w}$ is obtained from the largest eigenvector of $\boldsymbol{S}_w^{-1} \boldsymbol{S}_b$. Since $\boldsymbol{S}_w^{-1} \boldsymbol{S}_b$ has rank $C-1$, there are $C-1$ optima $\boldsymbol{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_{C-1}]$. By projecting data onto the space spanned by $\boldsymbol{W}$, we achieve dimensionality reduction and feature extraction of data onto the most discriminant subspace.

Kernel LDA [25, 26, 27] can be formulated by using the kernel trick as follows. Let $\Gamma : \mathbb{R}^d \to \mathcal{F}$ be a non-linear map from the input space $\mathbb{R}^d$ to a feature space $\mathcal{F}$, and $\Gamma = [\boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_N]$ be the feature matrix of the mapped training points $\boldsymbol{\gamma}_i$. Assuming $\boldsymbol{w}$ is a linear combination of those feature vectors, $\boldsymbol{w} = \boldsymbol{\Gamma}\boldsymbol{\alpha}$, we can use the kernel trick and rewrite the Rayleigh quotient in terms of $\boldsymbol{\alpha}$ as:

$$\begin{aligned} Ra(\boldsymbol{\alpha}) &= \frac{\boldsymbol{\alpha}^\top \boldsymbol{\Gamma}^\top \boldsymbol{S}_b \boldsymbol{\Gamma} \boldsymbol{\alpha}}{\boldsymbol{\alpha}^\top \boldsymbol{\Gamma}^\top \boldsymbol{S}_w \boldsymbol{\Gamma} \boldsymbol{\alpha}} = \\ &= \frac{\boldsymbol{\alpha}^\top \boldsymbol{K}(\boldsymbol{V} - \boldsymbol{e}_N \boldsymbol{e}_N^\top / N)\boldsymbol{K}\boldsymbol{\alpha}}{\boldsymbol{\alpha}^\top (\boldsymbol{K}(\boldsymbol{I}_N - \boldsymbol{V})\boldsymbol{K} + \sigma^2 \boldsymbol{I}_N)\boldsymbol{\alpha}} = \\ &= \frac{\boldsymbol{\alpha}^\top \boldsymbol{\Sigma}_b \boldsymbol{\alpha}}{\boldsymbol{\alpha}^\top (\boldsymbol{\Sigma}_w + \sigma^2 \boldsymbol{I}_N)\boldsymbol{\alpha}}, \end{aligned} \tag{6}$$

where $\boldsymbol{K}$ is the kernel matrix, $\boldsymbol{e}_N$ is a vector of ones that has length N, $\boldsymbol{V}$ is a block-diagonal matrix whose c-th block is the matrix $\boldsymbol{e}_{N_c} \boldsymbol{e}_{N_c}^\top / N_c$, and $\boldsymbol{\Sigma}_b = \boldsymbol{K}(\boldsymbol{V} -$

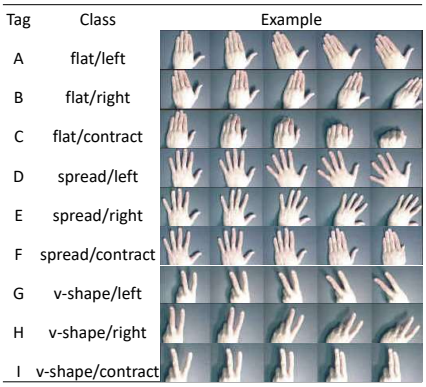


Figure 3: Examples of Cambridge hand gestures.

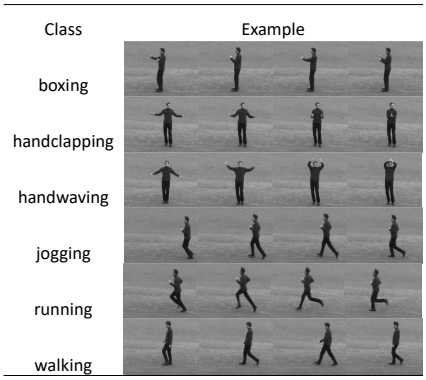


Figure 4: Examples of KTH actions.

$\boldsymbol{e}_N \boldsymbol{e}_N^\top / N)\boldsymbol{K}$. In our framework, the kernel matrix, $\boldsymbol{K}$, is calculated as the similarity matrix between training subspaces, where each element can be written in terms of the projection kernel as $k_p(\widetilde{\boldsymbol{Y}}_q, \widetilde{\boldsymbol{Y}}_w)$, where q is a row and w is a column of $\boldsymbol{K}$. The term $\sigma^2 \boldsymbol{I}_N$ is used for regularizing the covariance matrix $\boldsymbol{\Sigma}_w = \boldsymbol{K}(\boldsymbol{I}_N - \boldsymbol{V})\boldsymbol{K}$. It is composed of the covariance shrinkage factor $\sigma^2 > 0$, and the identity matrix $\boldsymbol{I}_N$ of size N. The set of optimal vectors $\boldsymbol{\alpha}$ are computed from the eigenvectors of $(\boldsymbol{\Sigma}_w + \sigma^2 I_N)^{-1}\boldsymbol{\Sigma}_b$. We apply the GDA algorithm to the projected subspaces $\widetilde{\boldsymbol{Y}}_i^c$. The GDA corresponds to steps 8 to 10 in Algorithm 1, and steps 5 to 7 in Algorithm 2.

4. Experiments

In this section, we discuss the validity of the proposed method through hand gesture and human action recognition tasks.

4.1. Experiment with Cambridge Hand Dataset

We conducted two types of experiments with the Cambridge hand gesture dataset [20]. This dataset contains 9 classes of hand gesture videos, each in 5 illumination scenarios, and 20 sample videos for each of the scenarios and classes. The number of frames of each video ranges from 37 to 119. In addition, in the experiments, all the images were resized to 12×16 pixels, and the grayscale pixel values compose the image features. As a result, an original feature vector $\boldsymbol{x}_l^{i,c}$ had dimension 12×16 ($d = 192$). The number of selected frames k to build one TE feature is fixed at $k = 15$, and as a result the dimension of a TE feature vector $\boldsymbol{s}_z^{i,c}$ is $d \times k = 192 \times 15 = 2880$. The number of TE features for each set is fixed to be $Z = 100$. Figure 3 shows examples of this dataset.

In the first experiment, we performed a qualitative experiment to aid in the visualization of the proposed method mechanism. We utilized three classes of hand gestures from the Cambridge dataset: flat/contract (C), spread/right (E) and spread/contract (F).

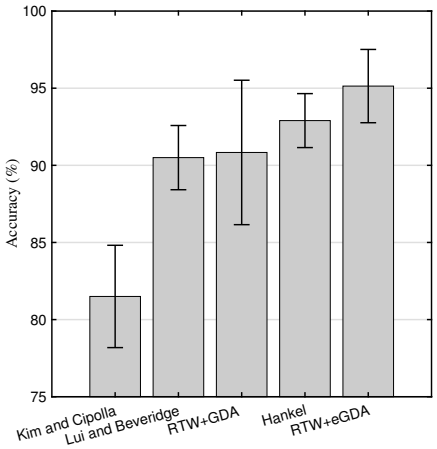


Figure 5: Results of the Cambridge Hand Dataset Experiment. The vertical axis refers to the average accuracy of all sets, for each of the methods in the horizontal axis. An error bar represents a method's standard deviation.

The normal illumination setting (set 5) was used for training, and the illumination setting of set 1 was used for testing. The parameters were set up in the following manner: dimension of hypo subspaces m was set to 7; dimension of class subspaces d_m was set to 50 ; and dimension of principal subspace d_p to 5.

In the second experiment, following the same setup as [2], we quantitatively compared our RTW+eGDA with the conventional methods: RTW+GDA, Kim and Cipolla [19], Lui [28] and Hankel [29]. These methods were selected as baselines due to their applications in motion representation and recognition. In Kim and Cipolla [19], the image sets of motions are described as linear subspaces, where a discriminative subspace is created in order to improve the feature extraction ability of the method. Lui [28] represents the image-sets of motions as a factorized tensor, where the geometry of the tensor space is extracted and compactly represented. Finally, in Hankel [29], image-sets of motions are described as autocorrelation matrices computed from Hankel trajectory matrices. In this approach, a discriminative subspace similar to [19] is employed to extract more useful features.

We used the 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 parameters were varied in the following manner: dimension of hypo subspaces m was varied from 5 to 7; dimension of class subspaces d_m was varied from 30 to 90 in increments of 20; and dimension of principal subspace d_p was varied from 5 to 30 in increments of 5. The results reported here are the best among the parameter settings.

The results of the quantitative evaluation can be seen in Figure 5. The vertical axis refers to the average accuracy of all sets. The error bars represent the method's standard deviation. We also conducted a t-test between RTW+eGDA and RTW+GDA with 4 samples and significance level $\alpha = 0.05$. From the test results, we can conclude with

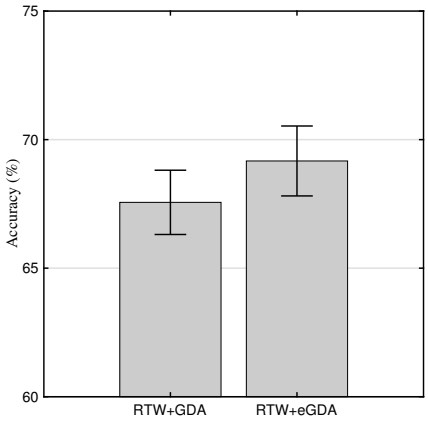


Figure 6: Results of the KTH Action dataset experiment. The vertical axis refers to the average accuracy of all 10 folds, for each of the methods in the horizontal axis. An error bar represents a method's standard deviation.

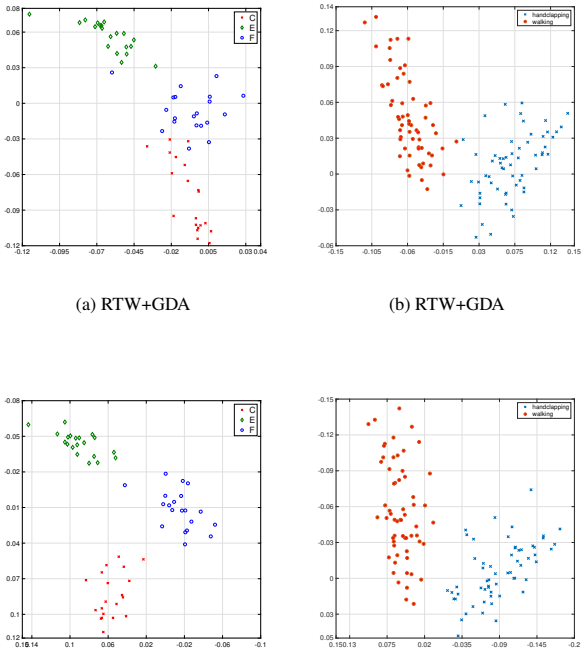
more than 95% confidence ($p = 0.0377$) that the proposed method performed better than the conventional method by using GDA.

4.2. Experiment with KTH Action Dataset

We also conducted experiments using the KTH action dataset [21]. Figure 4 shows examples of this database's 6 classes of actions, namely: boxing, hand clapping, hand waving, running, jogging, and walking. The dataset contains actions performed by 25 subjects in videos, filmed under 4 different shooting conditions: outdoors, outdoors with variation of zooming, outdoors with different clothes, and indoors.There are 4 sample videos for each of the conditions and classes. The number of frames of each video ranges from 37 to 119. In addition, in the experiments, all the images were resized to 16×16 pixels. In total there are 2391 sequences of actions.

In the first experiment, we performed a qualitative assessment. For each of the 6 classes, 10 subjects were randomly selected for training, and 15 for testing. The parameters were set up in the following manner: dimension of hypo subspaces m was set to 19; dimension of class subspaces d_m was set to 50 ; dimension of principal subspace d_p to 20; the number of selected frames k to build one TE feature is fixed at $k = 5$, and the number of TE features for each set is fixed to be $Z = 500$.

Figure 6 shows the results of the KTH Action Dataset Experiment. To quantitatively confirm the effectiveness of the orthogonalization of RTW class subspaces by GDS projection, we measured the Fisher criterion (class separability degree) among all the classes. The performance is higher as the separability degree approaches 1.0. Table 2 shows the experimental results. We can see that the separability degree of GDA is further improved by GDS projection.



(a) RTW+GDA (b) RTW+GDA

(c) RTW+eGDA (d) RTW+eGDA

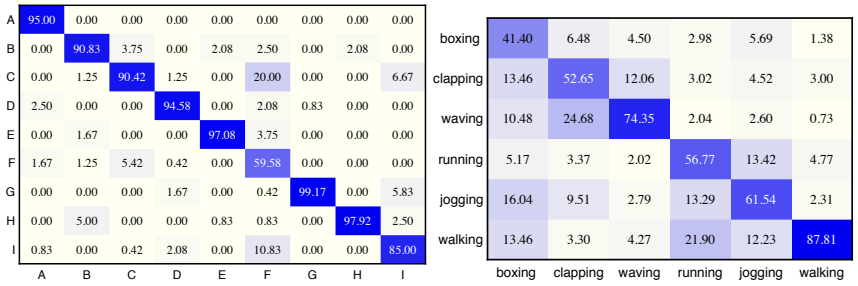
Figure 7: Scatter points of three Cambridge hand gesture classes by using RTW combined with (a) conventional GDA and (c) eGDA; and scatter points of two KTH action classes by using RTW combined with (b) conventional GDA and (d) eGDA.

Figure 7 shows scatter plots of the generated points corresponding to the test subspaces, which were generated from the 20 test sequences in each class. In this figure, (a) depicts the result of the combination of RTW and conventional GDA (RTW+GDA), and (c) shows the proposed method, RTW and the enhanced GDA (RTW+eGDA). The figure suggests that by using the proposed method, reduction of the distance between subspaces of the same class can be achieved. The scatter plots given by the Cambridge gesture dataset classes C (flat/contract), E (spread/right) and F (spread/contract) reveals visually that RTW+eGDA is able to produce higher discriminative features than RTW+GDA.

Figure 7 shows scatter plots of the generated points corresponding to the test sub-

Table 2: Separability of RTW+GDA and RTW+eGDA in the first experiment using the KTH dataset.

Method	Separability
RTW+GDA	0.23
RTW+eGDA	0.45



(a) RTW+GDA (average 90.83 %)

(b) RTW+GDA (average 67.56 %)

(c) RTW+eGDA (average 95.14 %)

(d) RTW+eGDA (average 69.17 %)

Figure 8: Confusion matrices of the (a) RTW+GDA in the Cambridge hand gesture experiment and (b) RTW+GDA in the KTH action experiment; (c) RTW+eGDA in the Cambridge hand gesture experiment and (d) RTW+eGDA in the KTH action experiment. The percentage in parentheses refers to the average accuracy of each method.

spaces of two classes: handwaving and running, as an example where two classes are comparatively well separated even by a two-dimensional discriminant space. In this figure, (b) depicts the result of the combination of RTW and conventional GDA (RTW+GDA), and (d) shows the proposed method, RTW and the enhanced GDA (RTW+eGDA).

In KTH dataset, the chosen classes for this plot have high overlap. We can observe that both RTW+GDA and RTW+eGDA produced very similar patterns. One observation regarding both investigated datasets is that in the Cambridge dataset temporal information plays an important role. On the other hand, in KTH dataset, temporal information seems to play a weaker role, since some classes (e.g. boxing, handclapping and hand waving) consist of iterations of the same action unit.

We graphically demonstrate that RTW+GDA and RTW+eGDA are well-suited for motion representation, even when dealing with complicated datasets containing cluttered and non-uniform backgrounds. These results encourage us to make another experiment and show the importance of combining RTW and eGDA for motion representation.

Figure 8 shows the confusion matrix of RTW+GDA (a) and RTW+eGDA (c). The

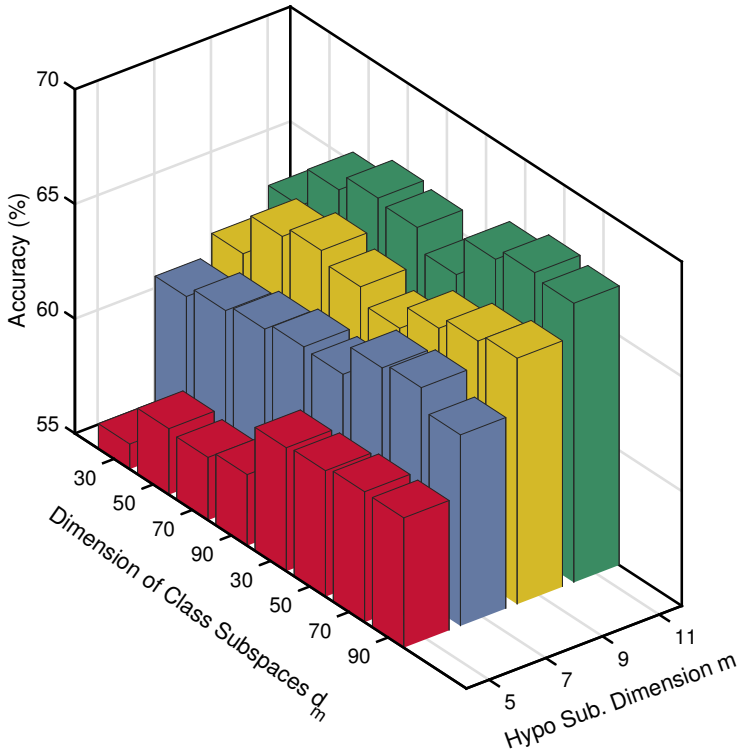


Figure 9: Parameter behavior of RTW+eGDA on the KTH action dataset.

vertical classes refer to predictions, while the horizontal classes refer to the ground truth. Each number represents the percentage of predictions attributed to a class in relation to their true class. The percentages between parenthesis in each matrix label refer to the average accuracy of the method.

Both RTW+GDA and RTW+eGDA provided efficient results at separating difficult classes in Cambridge dataset. For instance, the methods demonstrated high accuracy in overlapping classes such as A (flat/left), B (flat/right), D (spread/left) and E (spread/right). In the case of employing linear subspaces, where the intrinsic low dimension representation has high level of similarity, such results may weaken. When employing RTW, the temporal coherence between the ordered patterns is efficiently attained, producing very competitive results.

The class with the worst result was F (spread/contract). One of the reasons that can cause such a class to have a low accuracy compared to other classes is that this gesture has a very large number of structures in common with the other classes. Besides, when

GDS is employed, such common structures could be removed automatically.

In the second experiment, we quantitatively compared the methods by a 10-fold cross validation scheme, where in each fold, 10 subjects from the 25 were randomly selected for training. For each sequence, we used the bounding box from [30] 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 compared the combination of RTW and conventional GDA (RTW+GDA) with RTW and the enhanced GDA (RTW+eGDA). The number of selected frames k to build one TE feature is fixed at $k = 5$, and the number of TE features for each set is fixed to be $Z = 500$. The parameters were varied in the following manner: dimension of hypo subspaces m was varied from 5 to 20 in increments of 2; dimension of class subspaces d_m was varied from 30 to 90 in increments of 20; and dimension of principal subspace d_p was varied from 5 to 30 in increments of 5. Figure 9 shows the parameter behavior of RTW+eGDA on the KTH action dataset.

The results can be seen in Fig. 6. The vertical axis refers to the average accuracy of all 10 folds, for each method in the horizontal axis. The error bars represent the method's standard deviation. We again conducted a t-test between RTW+eGDA and RTW+GDA, this time with 10 samples and significance level $\alpha = 0.05$. From the test results, we can conclude with more than 95% confidence ($p = 0.007$) that the proposed method performed better than the conventional method by using GDA.

Figure 8 shows the confusion matrix of RTW+GDA (b), and RTW+eGDA (d). The vertical classes refer to predictions, while the horizontal classes refer to the ground truth. Each number represents the percentage of predictions attributed to a class in relation to their true class. The percentages between parenthesis in each matrix label refer to the average accuracy of the method.

From the confusion matrix of KTH dataset, we can observe that RTW+GDA and RTW+eGDA achieved competitive results, where RTW+eGDA substantially outperforms RTW+GDA in boxing, clapping and walking classes and achieved similar results in the remaining classes. In overall, RTW+eGDA outperforms RTW+GDA, justifying the applications of the proposed method.

The results obtained by RTW+eGDA for the classes running and jogging did not outperform RTW+GDA. This may be due to the limitations of representation based on linear subspaces. Specifically, the cause could be that the overlap rate between the two classes' distributions is very large, as linear subspaces are insufficient to represent the complicated boundary between these classes. In this particular case, the principal subspace may contain a substantial amount of data from both classes and when it is removed from the sum subspace, most of the representative information from both classes is also removed, leading to a degradation of both subspace classes, where its projections no longer can represent its semantics. We expect that the introduction of nonlinear kernels composited with Grassmann kernels would largely suppress misclassification due to this severe model overlap.

To elucidate on the effect of the introduced parameters on the proposed method's performance, we have performed an additional evaluation. Fig. 9 shows a bar plot of accuracy of the proposed RTW+eGDA, when fixing the dimension of principal

subspace d_p to 5, and varying both the dimension of hypo subspaces m and dimension of class subspaces d_m. In this dataset, 11 dimensional models have performed better, and small dimension m tends to induce a performance degradation due to a poor model representation. According to Fig. 9, the class subspace dimension d_m seems to depend on the dimension of hypo subspaces m. However, when $m = 11$, d_m is more invariant to small changes, so that no much emphasis needs to be put in searching an optimal value. Regarding dimension of the principal subspace, small values are usually best, to avoid losing substantial information that may be contained on the removed principal components.

4.3. Experiment with UCF Sports Dataset

We conducted a third experiment using the UCF sports dataset [22, 23]. The purpose of this experiment is twofold: to shed light on RTW+eGDA's potential with more challenging data; and to anticipate its behavior when using more sophisticated features. This experiment contrasts with the previous ones, which focused on assessing RTW+eGDA under the simplest scenario, by using raw images, elucidating its usefulness as a simple and practical means for further enhancing RTW+GDA.

The UCF sports dataset contains a total of 150 sequences of subjects performing sports, with 10 classes, namely: diving, golf swing, kicking, lifting, riding horse, running, skateboarding, swing-bench, swing-side, and walking. The number of frames of each video ranges from 50 to 70. The action bounding box has been extracted, using annotations provided. Then, each cropped image was resized to a 38×24 pixels grayscale image, resulting in a 912-dimensional vector for each frame.

We quantitatively compared the methods by a leave-one-out cross-validation scheme (LOOCV), a standard experiment setting for this data. That means 150 repetitions of learning, with one video as query and the remaining 149 videos as reference data.

To anticipate RTW+eGDA's potential when combined with feature extractors, we utilized pre-processing of each video frame by two features: a histogram of gradients (HOG), the combinations with which are named as HOG+RTW+GDA and HOG+RTW+eGDA; and convolutional neural network (CNN) features extracted from the last fully-connected layer of the AlexNet [31], the combinations with which we refer to as AlexNet+RTW+GDA/eGDA. The AlexNet was pre-trained on more than a million images from the ImageNet database [32], and has not been fine-tuned or equipped with mechanisms to represent the time components of the video data. We compared the above frameworks with RTW+GDA and RTW+eGDA.

In addition, we compare RTW+eGDA to a number of conventional methods that are relevant to the approach taken in the proposed method: 1) Motion extraction methods: robust non-linear knowledge transfer model (R-NKTM) and dense trajectory based method (DT), that have been recently proposed [33]. Namely, we compare 4 variants: trajectory DT, trajectory R-NKTM, HOG+HOF+MBH+Traj. DT and HOG+HOF+MBH+Traj. R-NKTM. Note that these methods are more elaborate in their feature extraction, especially the latter two that use an intricate combination of video descriptors HOG+HOF+MBH+Traj [34]. In contrast, our proposed method here uses only HOG and no feature fusion. 2) Methods for classification of image sets: Grassmann/subspace learning methods include discriminative canonical correlations (DCC) [19], constrained mutual subspace method (CMSM) [14], Grassmannian

graph-Embedding discriminant analysis (GGDA) [35] and projection metric learning (PML) [36]. We also compare a covariance-based method named covariance discriminant learning (CDL) [37]. We evaluate the performance of each method utilizing raw images, HOG and AlexNet features.

Regarding RTW parameters, the number of selected frames k to build one TE feature is fixed at $k = 3$, and the number of TE features for each set is fixed to be $Z = 60$. The HOG parameters were set as follows: number of bins is fixed at 9, the cell size at 5, and the block size at 3. The GDA and eGDA parameters were varied in the following manner: dimension of hypo subspaces m was varied from 8 to 14 in increments of 2; dimension of class subspaces d_m was varied either 40 or 50; and dimension of principal subspace d_p was varied from 1 to 8.

The results can be seen in Table 3. Both GDA and eGDA perform better with HOG and CNN features, indicating that using more discriminative features instead of raw images can improve them. It also can be noted that the performance gap between RTW+eGDA and RTW+GDA increased when using HOG. When compared to the conventional methods in this experiment, our method is competitive, with HOG+RTW+eGDA and AlexNet+RTW+eGDA achieving better results than methods using trajectory features and the baseline subspace-based based methods. These results demonstrate a potential for extensions and broad utility of the proposed method independent on the type of feature. In addition, our method would benefit from pre-trained deep neural networks, such as DenseNet and ResNet50. This potential is corroborated by the results shown by [38], which indicate that a subspace of deep features can be useful to represent image sets. Although the two methods based on a complicated combination of four types of features (HOG, HOF, MBH and trajectories) over-performed the proposed method, this result was expected to some extent, since these methods combine various types of motion analysis features intricately to obtain high performances.

RTW+eGDA overperforms CDL when using HOG features while the reverse happens when using AlexNet features. To verify if either method is overperforming the other meaningfully, we conducted a paired two-sample t-test between the results of RTW+eGDA and CDL with significance level 0.05 . From the test results ($p = 0.6187$), we cannot conclude with more than 95% confidence that both methods perform statistically significantly differently in this experiment. The reason for their comparable level may be that, as said in the CDL original paper, the covariance matrix is able to capture ordering in a set of patterns, which may work as a mechanism for representing the time structure like RTW. It also utilizes an LDA based classifier to predict categories. In the following we would like to discuss their differences further.

To demonstrate a situation where our proposed method should be considered as a good choice, we performed another experiment comparing the two best performing image-set based methods CDL and RTW+eGDA. Utilizing the AlexNet deep features we evaluate both frameworks in a situation of small sample size (SSS). Concretely, we have purposefully limited the number of frames available in an image sequence by a percentage of the total number of frames, from 20% to 50%. That is, for 20%, in a video with 100 frames, 20 frames would be used, selected by keeping one frame, skipping the next few frames, and then repeating that process until the sequence ends. The results can be seen in Table 4. As shown, the performance of CDL falls significantly as fewer frames are available. The reason is most likely that the rank of the covariance matrix

Table 3: Results of the UCF sports experiment. Results from DT and R-NKTM are reported in [33].

Methods	Accuracy (%)		
trajectory DT [33]	75.20		
trajectory R-NKTM [33]	76.70		
HOG+HOF+MBH+Traj. DT [33]	88.20		
HOG+HOF+MBH+Traj. R-NKTM [33]	88.20		
	Raw	HOG	AlexNet
DCC [19]	59.33	68.97	72.00
CMSM [14]	68.97	65.33	82.67
GGDA [35]	47.33	49.33	57.33
PML [36]	72.67	73.33	76.67
CDL [37]	70.00	76.00	86.00
RTW+GDA [2]	63.33	70.00	80.00
RTW+eGDA (proposed method)	70.00	78.00	84.67

tends to decrease causing instabilities when measuring distances, while subspaces in that context may offer a more robust model. Therefore, RTW+eGDA may be the method of choice in systems where the incoming stream of image data may have the drop of capture speed, or simply data is corrupted or few frames are available.

Another aspect of RTW+eGDA and CDL that is worth discussing is the necessary memory requirements to run the methods. RTW+eGDA requires memory of $dk \times m$ elements to store a subspace corresponding to one motion sequence. CDL utilizes $d \times d$ covariance matrices, which can be orders of magnitude higher than the memory required by the proposed method. For example, in the current experiment with HOG features, $dkm = 1458 \times 3 \times 8 = 34992$, while $d^2 = 1458^2 = 2125764$, meaning that RTW+eGDA uses only 1.6% of the memory used by CDL. Even if one exploits the symmetry of the matrices and reconstructs them from their upper/lower triangles at each time they are accessed in memory, one would still need $d(d+1)/2 = 1063611$ elements. Still RTW+eGDA uses only 3.3% of the memory used by CDL in this case, without the need to reconstruct the representation for storage. Therefore, RTW+eGDA can be one effective choice in mobile phones, car navigation, and embedded systems where memory is a limited resource.

Table 4: Performances in UCF sports when a small sample size (SSS) of videos frames is available.

Frames remaining (%)	20%	30%	40%	50%	100%
AlexNet+RTW+eGDA	76.67	78.00	80.67	80.00	84.67
AlexNet+CDL	57.33	67.33	76.67	82.67	86

5. Conclusions

In this paper we have proposed a combination of randomized time warping and eGDA, to address more effectively the classification of motion sequences. Our method may be used for various types of applications with continuous sequences, but in this

paper we focused on the applications of hand gestures and human action classification. The key idea of our enhanced Grassmann manifold is to project class subspaces onto a generalized difference subspace before mapping them on a Grassmann manifold. The GDS projection can extract the differences between classes and generate data points with optimized between-class separability on the manifold, which are more desirable for GDA. The validity of our enhanced Grassmann discriminant analysis was demonstrated through classification experiments with the Cambridge hand gesture, KTH action, and UCF sports datasets, where it outperformed its GDA counterpart and showed competitiveness with state-of-art methods. From the experiments we have also demonstrated that the proposed method can be a good choice in applications with small sample problems and those which require lower memory.

Future works include the introduction of multiple sophisticated features such as dense trajectories, and the investigation of their combination in terms of subspace representation. Besides, our framework would benefit from non-handcrafted features, where the descriptor is learned directly from data. Therefore, an attractive research line would be the investigation of features obtained from pre-trained deep neural networks, such as DenseNet and ResNet50.

Acknowledgements

We are grateful to the anonymous reviewers for their constructive comments, which improved the presentation and experiments in this paper. This work was partly supported by JSPS KAKENHI Grant Number 16H02842 and the Japanese Ministry of Education, Culture, Sports, Science and Technology (MEXT) scholarship.

References

[1] J. Hamm, D. D. Lee, Grassmann discriminant analysis: a unifying view on subspace-based learning, in: Proceedings of the 25th International Conference on Machine Learning, ACM, 2008, pp. 376–383.

[2] C. H. Suryanto, J.-H. Xue, K. Fukui, Randomized time warping for motion recognition, Image and Vision Computing 54 (2016) 1–11.

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

[4] O. Yamaguchi, K. Fukui, K. Maeda, Face recognition using temporal image sequence, Proc. International Conference on Automatic Face and Gesture Recognition (1998) 318–323.

[5] Y. Chikuse, Statistics on special manifolds, Vol. 174, Springer Science & Business Media, 2012.

[6] P. Turaga, A. Veeraraghavan, A. Srivastava, R. Chellappa, Statistical computations on Grassmann and Stiefel manifolds for image and video-based recognition, IEEE Trans. Pattern Analysis and Machine Intelligence 33 (11) (2011) 2273–2286.

[7] R. Slama, H. Wannous, M. Daoudi, A. Srivastava, Accurate 3D action recognition using learning on the Grassmann manifold, Pattern Recognition 48 (2) (2015) 556–567.

[8] T. Alashkar, B. B. Amor, M. Daoudi, S. Berretti, A Grassmann framework for 4D facial shape analysis, Pattern Recognition 57 (2016) 21–30.

[9] H. Tan, Z. Ma, S. Zhang, Z. Zhan, B. Zhang, C. Zhang, Grassmann manifold for nearest points image set classification, Pattern Recognition Letters 68 (2015) 190–196.

[10] J. Hamm, Subspace-based learning with Grassmann kernels.

[11] M. T. Harandi, C. Sanderson, S. Shirazi, B. C. Lovell, Kernel analysis on Grassmann manifolds for action recognition, Pattern Recognition Letters 34 (15) (2013) 1906–1915.

[12] K. Fukui, A. Maki, Difference subspace and its generalization for subspace-based methods, Pattern Analysis and Machine Intelligence, IEEE Transactions on 37 (11) (2015) 2164–2177.

[13] E. Oja, Subspace methods of pattern recognition, Vol. 6, Research Studies Press, 1983.

[14] K. Fukui, O. Yamaguchi, Face recognition using multi-viewpoint patterns for robot vision, in: Robotics Research. The Eleventh International Symposium, Springer, 2005, pp. 192–201.

[15] H. Tan, Y. Gao, Z. Ma, Regularized constraint subspace based method for image set classification, Pattern Recognition 76 (2018) 434–448.

[16] R. Zhu, K. Fukui, J.-H. Xue, Building a discriminatively ordered subspace on the generating matrix to classify high-dimensional spectral data, Information Sciences 382 (2017) 1–14.

[17] L. S. Souza, B. B. Gatto, K. Fukui, Enhancing discriminability of randomized time warping for motion recognition, in: Machine Vision Applications (MVA), Fifteenth IAPR International Conference on, IEEE, 2017, pp. 77–80.

[18] L. Souza, H. Hino, K. Fukui, 3D object recognition with enhanced Grassmann discriminant analysis, in: ACCV 2016 Workshop (HIS 2016), 2016.

[19] T.-K. Kim, J. Kittler, R. Cipolla, Discriminative learning and recognition of image set classes using canonical correlations, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (6) (2007) 1005–1018.

[20] T.-K. Kim, R. Cipolla, Canonical correlation analysis of video volume tensors for action categorization and detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (8) (2009) 1415–1428.

[21] C. Schuldt, I. Laptev, B. Caputo, Recognizing human actions: a local svm approach, in: Pattern Recognition, Proceedings of the 17th International Conference on, Vol. 3, IEEE, 2004, pp. 32–36.

[22] M. D. Rodriguez, J. Ahmed, M. Shah, Action mach a spatio-temporal maximum average correlation height filter for action recognition, in: Computer Vision and Pattern Recognition, IEEE Conference on, IEEE, 2008, pp. 1–8.

[23] K. Soomro, A. R. Zamir, Action recognition in realistic sports videos, in: Computer Vision in Sports, Springer, 2014, pp. 181–208.

[24] R. Fukunaga, Statistical pattern recognition.

[25] B. Scholkopft, K.-R. Mullert, Fisher discriminant analysis with kernels, Neural networks for signal processing IX 1 (1) (1999) 1.

[26] G. Baudat, F. Anouar, Generalized discriminant analysis using a kernel approach, Neural Computation 12 (10) (2000) 2385–2404.

[27] Y. Li, S. Gong, H. Liddell, Constructing structures of facial identities using kernel discriminant analysis, in: The 2nd International Workshop on Statistical and Computational Theories of Vision, 2001.

[28] Y. M. Lui, J. R. Beveridge, M. Kirby, Action classification on product manifolds, in: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on, IEEE, 2010, pp. 833–839.

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

[30] 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 (3) (2012) 533–547.

[31] A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet classification with deep convolutional neural networks, in: Advances in neural information processing systems, 2012, pp. 1097–1105.

[32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: Computer Vision and Pattern Recognition, IEEE Conference on, IEEE, 2009, pp. 248–255.

[33] H. Rahmani, A. Mian, M. Shah, Learning a deep model for human action recognition from novel viewpoints, IEEE Transactions on Pattern Analysis and Machine Intelligence 40 (3) (2018) 667–681.

[34] H. Wang, C. Schmid, Action recognition with improved trajectories, in: Computer Vision, IEEE International conference on, 2013, pp. 3551–3558.

[35] M. T. Harandi, C. Sanderson, S. Shirazi, B. C. Lovell, Graph embedding discriminant analysis on Grassmannian manifolds for improved image set matching, in: Computer Vision and Pattern Recognition, IEEE Conference on, IEEE, 2011, pp. 2705–2712.

[36] Z. Huang, R. Wang, S. Shan, X. Chen, Projection metric learning on Grassmann manifold with application to video based face recognition, in: Computer Vision and Pattern Recognition, IEEE Conference on, 2015, pp. 140–149.

[37] R. Wang, H. Guo, L. S. Davis, Q. Dai, Covariance discriminative learning: A natural and efficient approach to image set classification, in: Computer Vision and Pattern Recognition, IEEE Conference on, IEEE, 2012, pp. 2496–2503.

[38] N. Sogi, T. Nakayama, K. Fukui, A method based on convex cone model for image-set classification with cnn features, in: International Joint Conference on Neural Networks (IJCNN), 2018, pp. 1–8.