Yun and Gu *EURASIP Journal on Image and Video Processing* (2017) 2017:72
DOI 10.1186/s13640-017-0220-3

EURASIP Journal on Image and Video Processing

RESEARCH Open Access

Time-dependent bag of words on manifolds for geodesic-based classification of video activities towards assisted living and healthcare

Yixiao Yun* and Irene Yu-Hua Gu

Abstract

In this paper, we address the problem of classifying activities of daily living (ADL) in video. The basic idea of the proposed method is to treat each human activity in the video as a temporal sequence of points on a Riemannian manifold and classify such time series with a geodesic-based kernel. The main novelties of this paper are summarized as follows: (a) for each frame of a video, low-level features of body pose and human-object interaction are unified by a covariance matrix, i.e., a manifold point in the space of symmetric positive definite (SPD) matrices Sym^d_+; (b) a time-dependent bag-of-words (BoW+T) model is built, where its codebook is generated by clustering per-frame covariance matrices on Sym^d_+; (c) for each video, high-level BoW+T features are extracted from its corresponding sequence of per-frame covariance matrices; and (d) for activity classification, a positive definite kernel is formulated, taking into account the underlying geometry of our BoW+T features, i.e., the unit n-sphere. Experiments were conducted on two video datasets. The first dataset contains 8 activity classes with a total of 943 videos, and the second one contains 7 activity classes with a total of 224 videos. The proposed method achieved high accuracy (average 89.66%) and small false alarms (average 1.43%) on the first dataset. Comparison with six exisiting methods on the second dataset showed further evidence on the effectiveness of the proposed method.

Keywords: Activity of daily living (ADL), Riemannian manifolds, Time-dependent bag-of-words (BoW+T) model, Assisted living, Healthcare

1 Introduction

Video activity recognition is a trending topic and yet a challenging problem in the field of computer vision. The capability of automatically recognizing human activities is a key functionality of ambient intelligence. It has a wide range of applications, from surveillance in public and restricted areas, traffic safety, and sports analysis to assisted living and healthcare and many other social aspects. Among them, assisted living and healthcare have drawn increasing attention due to population ageing and the noticeable trend of independent living.

In this paper, we mainly focus on two aspects. First, we consider human activities that are typical in the context of assisted living and healthcare, where only several essential daily activities are handled, instead of a large number of general activities. More specifically, activities of interest in our case include activities of daily living (ADL) such as *eating* and *drinking* and anomalies like *falling down*. The purpose of studying ADL is to learn daily routines of individuals and to generate dedicated recommendations for a healthy living. As for anomalies, the aim is to trigger alarms when emergency occurs.

Secondly, we focus on video data and propose a video-based method for activity classification. Before that, we briefly review some existing and representative work on video activity recognition in the past few years. For example, Lin et al. modeled the entire scene as an error-free network, where each node corresponds to a patch of the scene and each edge represents the activity correlation between the corresponding patches. Based on this network, people

*Correspondence: yixiao1987@gmail.com
Department of Electrical Engineering, Chalmers University of Technology, SE-412 96 Gothenburg, Sweden

are modeled as packages and human activities are modeled as the process of package transmission [1]. Everts et al. recognized human actions in videos based on color spatio-temporal interest points (STIPs) [2] that are multichannel reformulations of STIP detectors and descriptors [3]. Zhang and Piccardi employed structural SVM for activity classification using spatio-temporal SIFT-based VLAD (vector of linearly aggregated descriptors) features [4]. Amer and Todorovic conducted activity recognition by representing activities using a sum-product network (SPN) [5]. Zhang and Parker introduced color-depth local spatio-temporal features for activity recognition based on orientation histograms in *xyzt* dimensions, where the histograms are built around interest points as local maxima of independent filters applied to different dimensions [6]. Recent years have also witnessed a significant advancement in various machine learning tasks using deep learning. Deep neural networks such as convolutional neural network (CNN) [7] and recurrent neural network (RNN) [8] have become common choices for image and video analysis, including the representation of video activities. For example, Baccouche et al. extended a CNN to 3D for learning spatio-temporal features and then trained an RNN to classify each sequence that contains human actions [9]. These methods mainly investigated spatio-temporal relations of human motions, where some promising results were reported. However, interacting objects as an important part of many activities with human-object interaction were paid less attention to. Further, activities as dynamic processes involving non-planar movement of human body lie on a nonlinear manifold, instead of vector space. This manifold nature was also under-explored.

In view of the issues mentioned above, we propose a novel method that jointly represents structural features for body pose and appearance features for interacting objects as a unified data point on a Riemannian manifold. By learning BoW features from this Riemannian manifold, we treat each video activity as a temporal sequence of manifold points (BoW features) on another Riemannian manifold. Then, we classify such time series with a kernel based on dynamic time warping (DTW) and geodesic distances.

More specifically, the main contributions of this paper include that (a) we use a unified covariance matrix to represent both structural and appearance features in each frame. These two different types of features correspond to body pose and human-object interaction, respectively. In this way, we obtain low-level features of each video as a temporal sequence of points on the Riemannian manifold of SPD matrices; (b) we build a BoW+T model on another Riemannian manifold, i.e., the unit n-sphere. The codebook of this model is learned by clustering per-frame covariance matrices from all videos in the training set. Considering the manifold structure of covariance matrices, geodesic distances and intrinsic means are used for the clustering; (c) we extract high-level features from the BoW+T model for each video as the final feature descriptor. It can be seen as a time series of points on the the unit n-sphere; (d) we formulate a positive definite kernel for activity classification using BoW+T features. This kernel is based on DTW and geodesic distances on the unit n-sphere.

The remainder of this paper is organized as follows: Section 2 briefly reviews the related work and theory. Section 3 gives an overview of the proposed method and then describes the major steps in detail. Section 4 shows experimental results on a video dataset containing activities from 8 classes. Finally, Section 5 concludes the paper.

2 Background information

In this paper, Riemannian manifolds are employed for feature representation of video activities. Therefore, in this section, we briefly review some theory and existing methods on Riemannian manifolds that are closely related to the proposed method, for the sake of mathematical and conceptual convenience in subsequent sections.

2.1 Riemannian geometry

Generally speaking, a manifold can be considered as a low-dimensional embedding in a high-dimensional space [10]. It represents the original data efficiently with lower dimensionality and still maintains key properties of the original data, such as topology and geometry. Manifolds are nonlinear structures that are not vector spaces; hence, Euclidean calculus does not apply. A Riemannian manifold is smooth and differentiable [10], where a set of metrics can be defined. In the tangent space of manifold points on a Riemannian manifold, linear operations may be performed.

2.1.1 The space of symmetric positive definite matrices

Mathematically, the space of $d \times d$ symmetric positive definite (SPD) matrices (Sym_+^d) is defined as

$$Sym_+^d = \bigcap_{\mathbf{x}\in\mathbb{R}^d} \left\{ \mathbf{P} \in Sym^d : \mathbf{x}^T\mathbf{P}\mathbf{x} > 0 \right\}, \tag{1}$$

which is an open convex cone, whose strict interior is a Riemannian manifold [10]. Two different metrics are commonly used to compute the statistics on Sym_+^d (Fig. 1), namely, the affine-invariant metric [11] and the log-Euclidean metric [12]. The log-Euclidean metric is used in this paper, as it has a closed-form solution and is computationally more efficient than affine-invariant metric [12]. Hence, below we only show equations under the log-Euclidean metric.

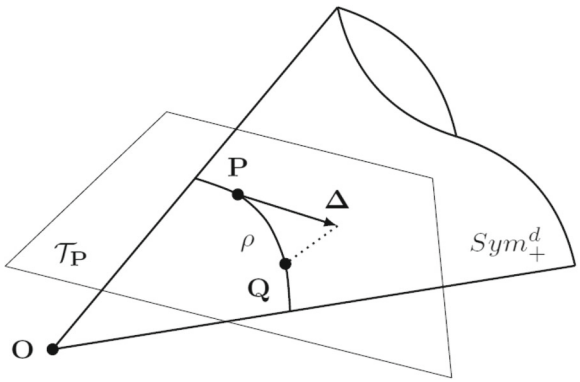

Fig. 1 Example of Sym_+^d ($d = 2$) embedded in a 3D space $\mathbb{R}^3$. **O** is the origin. **P** and **Q** are the manifold points, i.e., $\mathbf{P}, \mathbf{Q} \in Sym_+^d$. $\mathcal{T}_\mathbf{P}$ is the tangent space at **P**. $\Delta \in \mathcal{T}_\mathbf{P}$ is the tangent vector whose projected point on the manifold is **Q**. The geodesic ρ is the shortest curve between **P** and **Q** on the manifold

Two mapping functions, the *exponential map* and the *logarithm map*, are usually defined to switch between the manifold and tangent space at a given point. Under the log-Euclidean metric, the exponential map ($\exp_\mathbf{P}(\cdot) : \mathcal{T}_\mathbf{P} \mapsto Sym_+^d$) and the logarithmic map ($\log_\mathbf{P}(\cdot) : Sym_+^d \mapsto \mathcal{T}_\mathbf{P}$) are defined as [13]:

$$\exp_\mathbf{P}(\Delta) = \exp(\log(\mathbf{P}) + \Delta) = \mathbf{Q}; \tag{2}$$

$$\log_\mathbf{P}(\mathbf{Q}) = \log(\mathbf{Q}) - \log(\mathbf{P}) = \Delta, \tag{3}$$

where $\mathcal{T}_\mathbf{P}$ is the tangent space at a manifold point **P**, $\Delta \in \mathcal{T}_\mathbf{P}$ is the tangent vector whose projected point on the manifold is **Q**, $\exp(\cdot)$ is the matrix exponential, and $\log(\cdot)$ is the principal logarithm of a matrix defined as the inverse of the matrix exponential [12].

The *geodesic* is the shortest curve between two points on a manifold [14]. The geodesic distance, the length of the geodesic, is used to measure the distance between two manifold points. Under the log-Euclidean metric, the geodesic distance between **P** and **Q** on the manifold Sym_+^d is computed by [13]

$$\rho(\mathbf{P}, \mathbf{Q}) = \| \log_\mathbf{P}(\mathbf{Q}) \| = \| \log(\mathbf{Q}) - \log(\mathbf{P}) \|, \tag{4}$$

where $\| \cdot \|$ is the Frobenius norm.

The Riemannian geometry of Sym_+^d can be exploited when the extracted feature descriptors are covariance matrices, e.g., region covariance [15], since the SPD cone is exactly the set of non-singular covariance matrices [16].

2.1.2 The unit n-sphere

The unit n-sphere, $\mathcal{S}^n$, is an n-dimensional sphere with a unit radius, centered at the origin of $(n+1)$-dimensional Euclidean space. An intuitive example would be a unit circle ($n = 1$) in 2-D space, or a 2-D unit sphere ($n = 2$) in 3-D space. Mathematically, it is defined by

$$\mathcal{S}^n = \{\mathbf{p} \in \mathbb{R}^{n+1} : \|\mathbf{p}\| = 1\} \tag{5}$$

which can be considered as the simplest Riemannian manifold after the Euclidean space [17]. The geodesic distance between two manifold points $\mathbf{p}, \mathbf{q}$ on $\mathcal{S}^n$ is the great-circle distance (Fig. 2):

$$\rho(\mathbf{p}, \mathbf{q}) = \arccos(\mathbf{p}^T \mathbf{q}) \tag{6}$$

where $\arccos(\cdot) : [-1, 1] \to [0, \pi]$ is the inverse cosine function [18]. The great-circle distance between two manifold points is unique.

The Riemannian geometry of $\mathcal{S}^n$ can be utilized when the extracted feature vectors are normalized by the ℓ_2 norm, e.g., SIFT [19], HOG [20], LBP [21].

2.2 Bag-of-words model

The bag-of-words (BoW) model is originally used in document classification, where each document is considered as a bag of words and is represented as a vector of occurrence counts of words (a histogram over the vocabulary). This model has also been applied to image classification [22], treating each image as a document (a bag of visual words). The BoW representation of an image is obtained by first clustering a set of selected local image descriptors such as SIFT (usually with k-means clustering) to generate a visual vocabulary (or, codebook), followed by extracting a histogram by assigning each descriptor to its closest visual word.

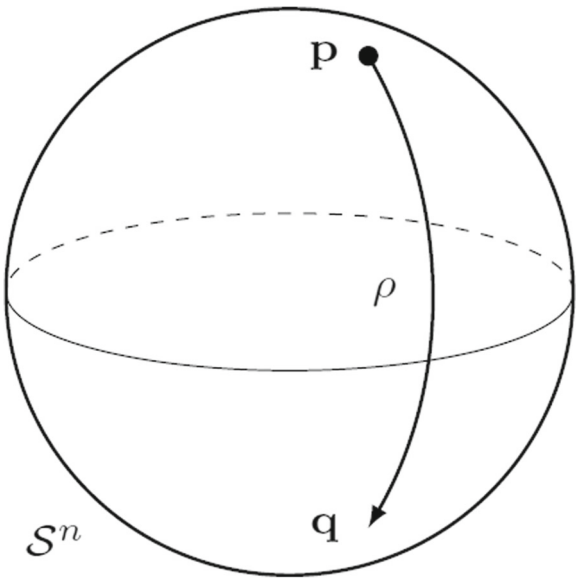

Fig. 2 Example of an n-sphere $\mathcal{S}^n$ ($n = 2$) embedded in an $(n+1)$-D space $\mathbb{R}^{n+1}$. $\mathbf{p}$ and $\mathbf{q}$ are manifold points, i.e., $\mathbf{p}, \mathbf{q} \in \mathcal{S}^n$. The geodesic ρ is the shortest curve between $\mathbf{p}$ and $\mathbf{q}$ on the manifold

The learning and recognition based on the BoW model can be roughly divided into two categories, namely, generative and discriminative models. Generative models estimate the probability of BoW features given a class, including Naïve Bayes classifier, and hierarchical Bayesian models such as probabilistic latent semantic analysis (pLSA) and latent Dirichlet allocation (LDA). Discriminative models learn a decision rule (classifier) to assign BoW representation of images to different classes, including nearest-neighbor classifier, SVM, AdaBoost, and kernel methods such as pyramid match kernel.

Since the BoW model is an orderless representation that counts frequencies of visual words from a dictionary, efforts have been made to incorporate spatial information into the model. For example, one can compute BoW features from sub-windows of the entire image, or based on part-based models [23]. Also, spatial pyramid representation is an extension of BoW features that gives locally orderless representation at several levels of resolution [24]. Moreover, the BoW model has been extended to encode higher-order statistics of the difference between visual words and pooled local features, such as Fisher Vectors (FV) [25] or vector of locally aggregated descriptors (VLAD) [26]. In this paper, we use the very basic BoW model other than its extensions, since we mainly focus on (i) building the model on manifold and (ii) adding temporal information into this model. Hence, a baseline approach would suffice our purpose as a proof of concept.

2.3 Distances and kernels for time series

A time series is an ordered finite set (a sequence) of data points, typically consisting of measurements observed successively over a time interval. Mathematically, it is defined as

$$Z = \{\mathbf{x}_t\}_{t=1}^{n} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\} \tag{7}$$

where $\mathbf{x}_t$ is the data point at time t and n is the total number of data points.

For time series classification, we first need to define a distance function $d(Z_i, Z_j)$ that measures the difference between each pair of time series Z_i and Z_j. Then, a kernel function $K(d(Z_i, Z_j))$ can be constructed, as a function of the distance function, to measure the similarity between each pair of time series Z_i and Z_j.

Some commonly used distance functions include dynamic time warping (DTW) [27], edit distance with real penalty (ERP) [28], and time warp edit distance (TWED) [29]. Kernel functions based on these distance measures have often been found to perform well in practice. However, they are not strictly positive definite, since DTW, ERP, and TWED in general are not positive definite [30].

Positive definiteness is a preferable property for kernel functions. It ensures that the optimization problem is convex and the solution is unique [31]. To this end, some positive definite kernels for time series classification have been suggested, e.g., global alignment (GA) kernels [32], recursive edit distance kernels (REDK) [30], which are shown to outperform indefinite kernels in general. In this paper, we propose to use a special type of REDK kernels in combination of a geodesic distance function, with the proof of its positive definiteness.

3 Proposed method for activity classification

This section first gives an overview of the proposed method, and then describes each important step of the method in details.

3.1 Overview of the proposed method

The proposed method can be summarized into three major steps, as depicted in Fig. 3. First of all, for each frame of a video activity, a unified covariance matrix is formed to jointly represent structural features of body pose and appearance features of interacting objects at hands. This covariance matrix can be viewed as a manifold point in the space of SPD matrices. Thus, a video activity is initially represented as a time sequence of covariance descriptors on the Riemannian manifold of SPD matrices. Then, a BoW model is learned by clustering the set of all covariance matrices from training video activities. For each video activity, time-dependent BoW features are extracted based on the learned BoW model. More specifically, a video activity is eventually characterized by a time series of BoW features. This time series can be seen as a trajectory on a unit n-sphere. Finally, a positive definite kernel is formulated based on DTW and geodesic distances on the unit n-sphere for activity classification, using these time-dependent BoW features.

3.2 Covariance descriptor for combining local appearance and global pose features

We adopt a part-based approach for feature extraction of a target person in each image frame, where the positions of left/right hand, head, feet, and torso axes of the person are required. The basic idea is to extract both appearance and structural features from body parts. The former may give important cues for local human-object interaction, while the latter can provide information on the global body pose and motion. These body parts can be detected by a *Kinect* sensor with skeleton tracking [33], or by existing toolboxes for pose estimation [34, 35]. The reason for detecting hand points is that interacting objects, as useful cues for activity recognition, are likely to appear in the vicinity of the human hands. It may be argued that hand regions are less important for activities without human-object interaction (e.g., *falling down*, *lying down*, *walking*, *sitting down*). However, they still provide useful information on arm movement relative to other body parts and serve as a discriminative feature between activities with

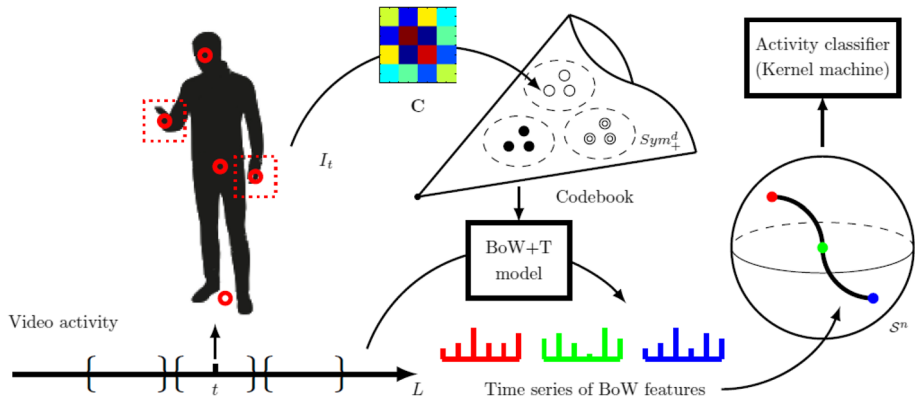

Fig. 3 Illustration of major steps in the proposed method. Notations and notes:
- I_t is the t-th frame of an input video, and L is the total number of frames
- "o" are key points (head, hands, waist center, midpoint of feet), and the areas with dotted edges are local patches centered at hands
- **C** is the frame-based covariance feature (as a point on the manifold of SPD matrices Sym_+^d) extracted from local patches and key points in I_t
- The codebook for BoW+T model is generated by clustering covariance matrices on Sym_+^d
- The video is encoded by the BoW+T model as a time series of manifold points on a unit n-sphere $\mathcal{S}^n$ and then classified by a kernel machine based on geodesic distance on that sphere

and without human-object interaction. It is also beneficial to detect the head, feet, and torso axes, as they may provide structural information about the body pose of the person.

For each image frame of a video activity containing a certain class of activity performed by a single person, a pair of 2-D hand points $\{\mathbf{p}_i\}$ are detected as $\mathbf{p}_i = (x_i, y_i)^T$, where $i = 1, 2$ is the hand index. For either hand point, a local image patch R_i of size $l \times l$ centered at $\mathbf{p}_i$ is obtained. For the j-th pixel in R_i, a feature vector $\mathbf{f}_{i,j}$ is formed by concatenating the following two component vectors:

a) Appearance feature vector [15, 16]

$$\mathbf{f}_{i,j}^a = \left[r, g, b, |I_x|, |I_y|, |I_{xx}|, |I_{yy}|, \sqrt{I_x^2 + I_y^2}, \arctan\left(\frac{I_y}{I_x}\right)\right]^T \tag{8}$$

where r, g, and b are the RGB values of the pixel, $|I_x|$, $|I_y|$, $|I_{xx}|$, and $|I_{yy}|$ are the magnitudes of the first and second derivatives along x, y directions, and $\sqrt{I_x^2 + I_y^2}$ and $\arctan\left(\frac{I_y}{I_x}\right)$ are the gradient magnitude and orientation, respectively.

b) Structural feature vector

$$\mathbf{f}_{i,j}^s = \left[x, y, \mathbf{d}_1^T, \mathbf{d}_2^T, \mathbf{d}_3^T, d_4\right]^T \tag{9}$$

where $(x, y)^T$ is the pixel coordinate in R_i, $\mathbf{d}_1$, $\mathbf{d}_2$, $\mathbf{d}_3$, and d_4 are the distances from the pixel to the head point $\mathbf{p}_a = (x_a, y_a)^T$, the other hand point $\mathbf{p}_k$ $(k \neq i, k = 1, 2)$, the midpoint $\mathbf{p}_b = (x_b, y_b)^T$ of two feet, and the torso axis, respectively. It is worth mentioning that (i) all these distances are normalized by the length of the torso axis L; (ii) $\mathbf{d}_1$, $\mathbf{d}_2$, and $\mathbf{d}_3$ are the 2-D vectors that contains distances in x and y directions; and (iii) d_4 is a scalar, i.e., distance from a point (the pixel) to a line (the torso axis).

Thus, the feature vector $\mathbf{f}_{i,j}$ for the j-th pixel in i-th local patch R_i related to the left (or right) hand is defined as

$$\mathbf{f}_{i,j} = \mathbf{\Omega}\left[\left(\mathbf{f}_{i,j}^a\right)^T, \left(\mathbf{f}_{i,j}^s\right)^T\right]^T \tag{10}$$

where $\mathbf{f}_{i,j}^a$ and $\mathbf{f}_{i,j}^s$ are feature vectors in (8) and (9) encoding local appearance of the interacting object and global pose of the target person, respectively, and $\mathbf{\Omega} > 0$ is an empirically determined diagonal matrix that adjusts the weight of features.

The local patch R_i for the i-th hand is represented by an $r \times r$ covariance matrix as

$$\mathbf{C}_i = \frac{1}{|R_i| - 1}\sum_{j=1}^{|R_i|} \tilde{\mathbf{f}}_{i,j}\tilde{\mathbf{f}}_{i,j}^T \quad \in Sym_r^+ \tag{11}$$

where $|R_i|$ is the total number of pixels in patch region R_i and $\tilde{\mathbf{f}}_{i,j}$ is the mean-subtracted feature vector.

Finally, assuming image patches at two hands are statistically independent, a covariance matrix of $d \times d$ $(d = 2r)$

is formed for each frame by using the local patch-based descriptors as follows:

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_{i^*} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_n \end{bmatrix} \quad \in Sym_d^+ \tag{12}$$

where $\mathbf{C}_{i^*}$ and $\mathbf{C}_n$ are computed from (11), and their indices (subscripts) $i^* = \arg\min_{i\in\{1,2\}} \|\mathbf{p}_a - \mathbf{p}_i\|$, and $n \neq i^*, n \in \{1,2\}$. Since covariance matrix $\mathbf{C} \in Sym_+^d$, it may be viewed as a point on a Riemannian manifold [16].

In this way, for each image frame of the video activity, the local appearance information of two hand regions which may potentially contain interacting objects and the global posture information as hand positions with respect to the head, feet, and torso axes are encoded into a unified covariance matrix, disregarding whether the person is left-handed or right-handed.

3.3 Temporal BoW model on the Riemannian manifold of SPD matrices

We employ the bag-of-words model for representing activities in videos. Since each image frame of a video activity is represented by a covariance descriptor (see Section 3.2), a most straightforward way would be to directly treat the video activity as a bag of covariance descriptors. However, temporal information as an important cue for activity recognition is neglected, which may lead to inferior results. Instead, in our case, each video activity is treated as a temporal sequence (time series) of bags of covariance descriptors. Further, comparing to representing the video activity as a time series of covariance matrices, the BoW model is more efficient and has been shown to be effective in many classification tasks. We refer to this temporal BoW model on Riemannian manifold as *Riemannian BoW+T model*.

The motivations for exploiting Riemannian manifolds in feature representation are threefolds: first, the nonlinear nature of manifolds enables effective description of dynamic processes of human activities involving nonplanar movement, which lie on a nonlinear manifold other than a vector space; secondly, many video features of human activities may be effectively described by low-dimensional data points on the Riemannian manifold while still maintaining the important property of human activities such as topology and geometry; thirdly, the Riemannian geometry provides a way to measure the distances of different activities on the nonlinear manifold, hence is suitable tool for the classification.

Given a set of covariance descriptors (manifold points) $\mathcal{X} = \{\mathbf{X}_i\}_{i=1}^M$, $\mathbf{X}_i \in Sym_d^+$, extracted and collected from a training set of video activities, we aim to learn a codebook (or, a dictionary) for our BoW model. In the simplest case, one can ignore the Riemannian geometry of SPD matrices and learn a codebook straight from the vectorized form of these matrices. That is, Euclidean geometry is applied and arithmetic mean is used for computing the clusters. Despite the simplicity, this method often yields undesirable outcome due to the *swelling effect* [1] [12]. Hence, the underlying Riemannian geometry should be taken into account for creating the codebook without swelling effect [2]. One common alternative is to first project the set of manifold points to a global tangent space at a particular point on the manifold and then apply Euclidean tools for clustering. However, mapping data to a tangent space only produces a first-order approximation of the data that can be distorted, especially in regions far from the origin of the tangent space. Therefore, we propose to use the intrinsic mean for obtaining the codebook, by extending k-means clustering to the case of Riemannian manifolds with the *Karcher mean* (also known as the Fráchet or Riemannian mean).

In this case, we aim to partition the set of M manifold points into k ($k \leq M$) subsets (or, clusters) $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \cdots, \mathcal{C}_k\}$ by minimizing the sum of squared geodesic distances of each manifold point in the cluster to its center. The objective is to seek:

$$\arg\min_{\mathcal{C}} \sum_{j=1}^{k} \sum_{\mathbf{X}_i \in \mathcal{C}_j} \rho^2(\boldsymbol{\mu}_j, \mathbf{X}_i) \tag{13}$$

where $\rho(\cdot,\cdot)$ is the geodesic distance defined in (4) and $\boldsymbol{\mu}_j \in Sym_d^+$ is the Karcher mean of points in the j-th cluster, which is found by

$$\arg\min_{\boldsymbol{\mu}_j} \sum_{\mathbf{X}_i \in \mathcal{C}_j} w_i \rho^2(\boldsymbol{\mu}_j, \mathbf{X}_i) \tag{14}$$

where $w_i \in \mathbb{R}$ is the weight for the i-th point that is inversely proportional to the distance from the point to its cluster center. The minimization problem in (14) can be solved by iteratively mapping from manifold to tangent spaces and vice versa until convergence [13]:

$$\boldsymbol{\mu}_j^{m+1} = \exp_{\boldsymbol{\mu}_j^m}\left(\sum_{\mathbf{X}_i \in \mathcal{C}_j} w_i \log_{\boldsymbol{\mu}_j^m}(\mathbf{X}_i) / \sum_{\mathbf{X}_i \in \mathcal{C}_j} w_i\right) \tag{15}$$

where $\exp_{(\cdot)}(\cdot)$ and $\log_{(\cdot)}(\cdot)$ are the pair of exponential and logarithm mapping functions defined in (2) and (3) under log-Euclidean metric and m is the index for the current iteration. Although one may argue that the iterative approach in (15) is computationally expensive, it shows superior performance comparing to extrinsic methods in our experiment.

Given a codebook of covariance descriptors $\{\boldsymbol{\mu}_j\}_{j=1}^k$ that is learned from (13), each video activity as a time series of covariance matrices $\{\mathbf{C}_t\}_{t=1}^L$, where $\mathbf{C}_t \in Sym_d^+$ and L is the length of the video activity (number of frames) can be encoded by the Riemannian BoW+T model as a time series of bags of covariance descriptors as follows.

1. Assign each covariance descriptor to its closest vocabulary word in the dictionary according to the geodesic distance in (4):
$$v_t = \arg \min_{j\in\{1,2,\cdots,k\}} \rho(\boldsymbol{\mu}_j, \mathbf{C}_t) \tag{16}$$
2. Temporally divide the video activity into N (fixed) segments $\{\mathcal{Z}_i\}_{i=1}^N$, each of length $\lfloor L/N \rfloor$. If $L < N$, then the segment length is chosen as 1, and the number of segments becomes L.
3. For each segment, generate a histogram $\mathbf{h}_i$ with k bins, and set the j-th bin to
$$c_{ij} = \sum_{t:\mathbf{C}_t\in\mathcal{Z}_i} \mathbb{I}\left[v_t = j\right] \tag{17}$$
where c_{ij} denotes the count of covariance descriptors that belong to the i-th segment $\mathcal{Z}_i$ and are assigned to the j-th codeword (cluster), and $\mathbb{I}[A]$ is an indicator function which equals 1 if event A is true, and 0 otherwise.
4. Normalize each histogram $\mathbf{h}_i$ by ℓ_2 norm.

In this way, each video activity is represented as a time series of histograms $\{\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_l\}$ over the vocabulary learned from (13), where each histogram $\mathbf{h}_i$ is a BoW feature vector based on covariance descriptors. Since all histograms are ℓ_2-normed, the Riemannian geometry of $\mathcal{S}^n$ can be utilized where video activities can also be viewed as temporal sequences of manifold points on a unit n-sphere $\mathcal{S}^n$, for some n.

Alternatively, one can temporally divide each video activity into segments with a fixed time interval and generate BoW feature vector from each segment in a same way. However, this may not be suitable for datasets containing video activities with significantly different length, especially for activities from the same class.

3.4 Time series classification with regularized DTW kernel based on geodesic distances on $\mathcal{S}^n$

For each pair of video activities, we need to measure the similarity between them. This is done by representing each video activity as a time series of manifold points on a unit n-sphere by the Riemannian BoW+T model (see Section 3.3) and comparing them with a distance measure. The essence for using DTW-based kernels and geodesic distance-based local kernels is to fit for two important aspects of the classification problem: (i) the sequential nature of our feature data points; (ii) the underlying nonlinear manifold structure of data sequence.

As aforementioned, dynamic time warping is a common way for the comparison between time series. However, DTW kernel is in general not positive definite, which may lead to inferior results. Instead, we employ a regularized version of DTW kernel that can be positive definite if certain conditions are satisfied. For detailed expression of this regularized DTW kernel, readers are referred to [30]. In fact, this regularized DTW kernel is a special type of REDK kernels [30], whose definiteness will be elaborated in a theorem in the Appendix of this paper.

Moreover, considering the underlying geometry of given time series, we propose to use a local kernel that is based on geodesic distances between manifold points on a unit n-sphere $\mathcal{S}^n$ in (6). More specifically, the local kernel is defined as
$$\begin{aligned} k(\mathbf{x}, \mathbf{y}) &= \exp\left(-\gamma\rho(\mathbf{x}, \mathbf{y})\right) \\ &= \exp\left(-\gamma \arccos\left(\mathbf{x}^T\mathbf{y}\right)\right), \end{aligned} \tag{18}$$
where γ is a stiffness parameter that weights the contribution of the local elementary costs. For detailed proof regarding the positive definiteness of the proposed kernel, please refer to the Appendix of this paper.

4 Experimental results

This section describes the experiments and shows the results on two video datasets containing activities from multiple classes using the proposed method.

4.1 Video datasets on activity classification

Dataset-A: This video dataset contains a total of 943 video activities from 8 activity classes, namely, (1) *eating*, (2) *drinking*, (3) *using laptop*, (4) *reading*, (5) *falling down*, (6) *lying down*, (7) *walking*, and (8) *sitting down*. The videos were recorded by ourselves at Chalmers University of Technology, Gothenburg, Sweden, using a Kinect™ sensor. There are 34 participants involved to increase the randomness in performing activities, without any pretraining. The frame rate is 30 frames per second. The frame resolution is 640×480. The average length of video is approximately $100 \sim 600$ frames ($\approx 3 \sim 20$ s). Detailed information on this dataset is provided in Table 1. As shown in Table 1, activities from different classes take up comparable proportions. Figure 4 depicts some key frames of the videos from *Dataset-A*.

Table 1 Specifications on *Dataset-A*

Class no.	Activity	No. of videos	Duration of videos (no. of frames [min, max])
1	*Eating*	108	[272, 1376]
2	*Drinking*	108	[256, 944]
3	*Using laptop*	105	[240, 1160]
4	*Reading*	105	[72, 1184]
5	*Falling down*	109	[6, 30]
6	*Lying down*	107	[10, 76]
7	*Walking*	150	[5, 63]
8	*Sitting down*	151	[7, 40]

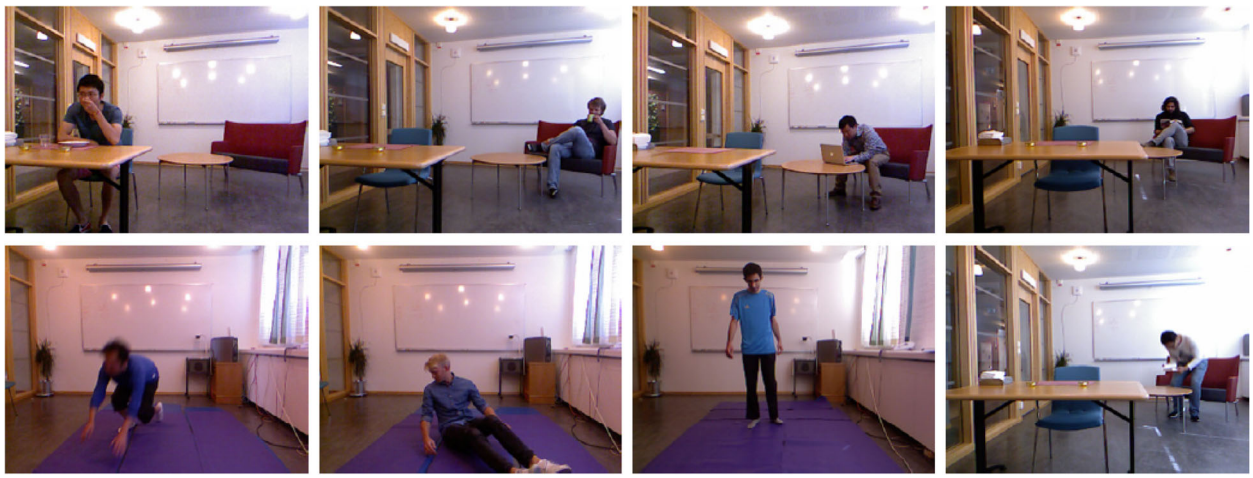

Fig. 4 Key frames from *Dataset-A* containing activities from 8 classes. Upper row from left to right: *eating*, *drinking*, *using laptop*, and *reading*. Lower row from left to right: *falling down*, *lying down*, *walking*, and *sitting down*

Dataset-B: This video dataset [36] contains a total of 224 RGB-D videos from 7 activity classes, namely, (1) *drinking*, (2) *eating*, (3) *using laptop*, (4) *reading cellphone*, (5) *making phonecall*, (6) *reading book*, and (7) *using remote*. The videos are captured by a Kinect™ sensor. There are 16 participants involved to perform each class of activity. The frame rate is 30 frames per second. The frame resolution is 640×480. The average length of video is approximately $260 \sim 530$ frames ($\approx 8 \sim 17$ s). Detailed information on this dataset is provided in Table 2. As shown in Table 2, activities from different classes take up exactly the same proportions. Figure 5 depicts some key frames of the videos from *Dataset-B*.

4.2 Experimental setup

For adjusting the weight of features, the diagonal matrix in (10) is $\mathbf{\Omega} = \text{diag}(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 4)$. For the BoW model, the number of codewords (clusters) $k = 150$. For each video activity, the number of segments $M = 7$. These parameters are empirically determined, without much tuning or optimization.

To limit the impact of inaccurate skeleton/pose estimation, we use manually marked key points in our tests for *Dataset-A*. However, for *Dataset-B*, key points are taken from skeletal joints that are automatically estimated by Kinect™, to ensure the fairness in comparison with other methods.

The libSVM [37] software was modified by using the proposed kernel to fit for our classifier, with the regularization coefficient and kernel parameters tuned by coarse-to-fine grid search and cross-validation. For both datasets, the classifiers were trained on approx. 50% videos from each class, and the remaining ones (approx. 50%) were used for testing.

Table 2 Specifications on *Dataset-B*

Class no.	Activity	No. of videos	Duration of videos (no. of frames [min, max])
1	*Drinking*	32	[228, 747]
2	*Eating*	32	[329, 598]
3	*Using laptop*	32	[210, 342]
4	*Reading cellphone*	32	[262, 423]
5	*Making phonecall*	32	[297, 579]
6	*Reading book*	32	[297, 481]
7	*Using remote*	32	[211, 544]

4.3 Tests, evaluations, and comparisons

The proposed method is tested on both datasets, with evaluations as well as comparisons to other methods where applicable.

4.3.1 Results on Dataset-A

For *Dataset-A*, the confusion matrix for the proposed method on the test set is given in Table 3. The performance of the proposed method in terms of classification accuracy and false positive rate (FPR) on the test set are reported in Table 4. It can be observed from Tables 3 and 4 that the proposed method overall achieved high classification accuracy and low false positive rate. Confusions are found between *eating*/*drinking*/*using laptop*/*reading* and *walking*/*sitting down* which may appear to be unusual. This is probably due to the fact that in some *walking* or *sitting down* scenarios, the person is holding something

Fig. 5 Key frames from *Dataset-B* containing activities from 7 classes. Upper row from left to right: *drinking*, *eating*, *using laptop*, and *reading cellphone*. Lower row from left to right: *making phonecall*, *reading book*, and *using remote*

(e.g., food, drink, book, laptop) while walking or sitting down.

4.3.2 Results on Dataset-B

For *Dataset-B*, the performance of the proposed method and several existing methods[3] in terms of classification accuracy on the test set is reported in Table 5. Below, we briefly summarize these methods that are compared with:

- Orderlet + Boosting/SVM [36] integrates three types of features to construct a spatio-temporal representation, including pairwise joint distances, spatial joint coordinates, and temporal variations of joint locations.
- Actionlet Ensemble [38] defines an actionlet as a particular conjunction of features for a subset of skeleton joints, indicating a structure of the features. Based on it, one human action can be interpreted as an actionlet ensemble that is a linear combination of the actionlets.
- DSTIP + DCSF [39] extends STIP to depth video as DSTIP and extracts depth cuboid similarity feature (DCSF) to describe the local 3-D depth cuboid around DSTIPs for activity recognition.
- EigenJoints [40] proposes a dimension-reduced skeleton feature, by using the spatial position differences between detected joints as well as the temporal differences between corresponding joints.
- Moving Pose [41] proposes a moving pose descriptor for capturing dynamic postures, by using the configuration, speed, and acceleration of the skeleton joints.

It can be observed from Table 5 that the proposed method achieved the highest classification accuracy, providing further evidence for the effectiveness of the

Table 3 Confusion matrix for the test set of *Dataset-A*

		Predicted class								Accuracy (%)
		Eating	*Drinking*	*Using laptop*	*Reading*	*Falling down*	*Lying down*	*Walking*	*Sitting down*	
True class	*Eating*	49	2	0	0	0	0	0	3	90.74
	Drinking	3	46	1	3	0	0	0	1	85.19
	Using laptop	1	1	45	3	0	0	0	2	86.54
	Reading	0	2	4	46	0	0	0	0	88.46
	Falling down	0	0	0	0	50	4	0	0	92.59
	Lying down	0	0	0	0	4	49	0	0	92.45
	Walking	1	0	3	4	0	0	67	0	89.33
	Sitting down	2	0	2	2	0	0	0	69	92.00

Table 4 Performance of the proposed method on activity classification (8 classes) using *Dataset-A*: classification accuracy, and false positive rate (FPR) on the test set

	Accuracy (%)	FPR (%)
Eating	90.74	1.67
Drinking	85.19	1.18
Using laptop	86.54	2.36
Reading	88.46	2.84
Falling down	92.59	0.95
Lying down	92.45	0.95
Walking	89.33	0
Sitting down	92.00	1.50
Overall (*)	89.77	–
Average	89.66	1.43

(*) Overall: the total number of true positives for all classes divided by the total number of videos in the test set

proposed method. Also, it is worth noting the performance drop on *Dataset-B*, comparing to *Dataset-A*. This is probably due to the fact that key points used for experiments on *Dataset-B* are automatically estimated by Kinect™, which may be less accurate than manually marking.

4.4 Discussions

The proposed method is shown to have better performance than other methods on *Dataset-B*. This is probably due to the following major differences between the proposed method and the other ones: (i) instead of joint representation of features through concatenation, we compute the covariance matrix of these features and use it as the low-level feature descriptor. The covariance descriptor encodes information of the variances of the defined features, and their correlations with each other.

Table 5 Performance of different methods on activity classification (7 classes) using *Dataset-B*: classification accuracy on the test set

Method	Accuracy (%)
Orderlet + Boosting [36]	71.4
Orderlet + SVM [36]	68.7
Actionlet Ensemble [38]	66.0
DSTIP + DCSF [39]	61.7
EigenJoints [40]	49.1
Moving Pose [41]	38.4
Proposed	72.34

Comparing to feature concatenation, covariance descriptor is a much more compact, efficient, and effective representation; (ii) in addition to spatio-temporal information that is exploited in other methods, we also consider local appearance information that encodes human-object interactions; (iii) other than the Euclidean metrics that is adopted in other methods, we take into account the underlying manifold geometry of the feature data points for classification.

For *Dataset-A*, to limit the impact of inaccurate skeleton/pose estimation on the proposed method, we used manually marked key points in our tests. Hence, when being replaced by automatically detected key points, some performance degradation is expected, if the key points on the skeleton are less accurate. This is also a possible reason of the performance drop on *Dataset-B*, comparing to *Dataset-A*. Although there are many toolboxes that can be exploited, such as [34, 35], the study of the impact of inaccurate skeleton/pose estimation on activity classification is beyond the scope of this paper.

5 Conclusion

In this paper, we proposed a method on human activity classification in video that is dedicated to assisted living and healthcare. The method treats each video activity as a temporal sequence of BoW features on a Riemannian manifold and classifies such time series with a kernel based on dynamic time warping (DTW) and geodesic distances. Experiments were conducted on two video datasets containing a total number of 943 videos from 8 classes and 224 videos from 7 classes, respectively. The proposed method achieved high classification accuracy and small false alarms overall, as well as for each individual class. Comparison with several existing methods provided further evidence for the effectiveness of the proposed method.

Endnotes

[1] A consequence from the Euclidean averaging of SPD matrices: the determinant of the Euclidean mean can be strictly larger than the original determinants, which is physically unacceptable.

[2] The determinant of a mean of SPD matrices remains bounded by the values of the determinants of the averaged matrices.

[3] The results of all these methods have been originally reported in [36].

Appendix

To show the positive definiteness of the regularized DTW kernel (REDK) with the local kernel of our choice in (18), we start with the following theorem.

Theorem 1 (Definiteness of REDK [30]) *Let $\mathbb{U}$ be the set of finite sequences (time series) and Ω be the empty sequence (with null length). REDK is a positive definite kernel on $\mathbb{U} \times \mathbb{U}$ if the local kernel $k(\mathbf{x}, \mathbf{y}) = f(\Gamma(\mathbf{x} \to \mathbf{y}))$ is a positive definite kernel on $((\mathcal{S} \times \mathcal{T}) \cup \{\Omega\})^2$, where $\mathcal{S}$ embeds the multidimensional space variables and $\mathcal{T} \subset \mathbb{R}$ embeds the time stamp variable, $\mathbf{x} \to \mathbf{y}$ is an edit operation on a pair $(\mathbf{x}, \mathbf{y}) \in ((\mathcal{S} \times \mathcal{T}) \cup \{\Omega\})^2$, and $\Gamma(\mathbf{x} \to \mathbf{y})$ is the associated cost (or, distance) function.*

From the above theorem, we know that it is the positive definiteness of our local kernel in (18) that matters, which leads us to the theorem below.

Theorem 2 (Schoenberg's Theorem [42, 43]) *Let $\mathcal{X}$ be a nonempty set and $f : (\mathcal{X} \times \mathcal{X}) \to \mathbb{R}$ be a kernel. The kernel $\exp(-\gamma f(\mathbf{x}, \mathbf{y}))$ is positive definite for all $\gamma > 0$ if and only if f is conditionally negative definite.*

Therefore, we need to show that the pairwise geodesic distance function $\rho(\mathbf{x}, \mathbf{y})$ (or, the inverse cosine function $\arccos(\mathbf{x}^T\mathbf{y})$) in (18) itself as a kernel f is conditionally negative definite. First of all, the definition of conditionally negative definite kernels is given as follows.

Definition 1 (Conditionally Negative Definite Kernels [44]) *A kernel $f : (\mathcal{X} \times \mathcal{X}) \to \mathbb{R}$ is called (conditionally) negative definite if it it symmetric and $\sum_{i,j=1}^{m} c_i c_j f(\mathbf{x}_i, \mathbf{x}_j) \leq 0$ for all $m \in \mathbb{N}$, $\{\mathbf{x}_1, \cdots, \mathbf{x}_m\} \subseteq \mathcal{X}$ and $\{c_1, \cdots, c_m\} \subseteq \mathbb{R}$ with $\sum_{i=1}^{m} c_i = 0$.*

Then, we recall the Taylor series of the inverse cosine function

$$\arccos(z) = \frac{\pi}{2} - \sum_{n=0}^{\infty} \left(\frac{(2n)!}{2^{2n}(n!)^2} \right) \frac{z^{2n+1}}{(2n+1)}. \tag{19}$$

From this series, it is clear that $\arccos\left(\mathbf{x}^T\mathbf{y}\right)$ is conditionally negative definite, because it is of the form "constant minus positive definite" [44]. For detailed proof, observe that with the above power series representation in (19), we have

$$\begin{aligned} f(\mathbf{x}_i, \mathbf{x}_j) &= \arccos\left(\mathbf{x}_i^T\mathbf{x}_j\right) \\ &= \frac{\pi}{2} - \sum_{n=0}^{\infty} \left(\frac{(2n)!}{2^{2n}(n!)^2} \right) \frac{\left(\mathbf{x}_i^T\mathbf{x}_j\right)^{2n+1}}{(2n+1)} \\ &= \frac{\pi}{2} - h(\mathbf{x}_i, \mathbf{x}_j), \end{aligned} \tag{20}$$

where $h(\mathbf{x}_i, \mathbf{x}_j)$ is a positive definite kernel. To see this, observe that the power series in (20) has nonnegative coefficients, and since $(\mathbf{x}_i^T\mathbf{x}_j)^{2n+1}$ is point-wise product of kernels, it is itself a kernel. Thus, we have in particular that the matrix

$$\begin{aligned} \mathbf{F} &= \begin{bmatrix} f(\mathbf{x}_1, \mathbf{x}_1) & f(\mathbf{x}_1, \mathbf{x}_2) & \cdots & f(\mathbf{x}_1, \mathbf{x}_m) \\ f(\mathbf{x}_2, \mathbf{x}_1) & f(\mathbf{x}_2, \mathbf{x}_2) & \cdots & f(\mathbf{x}_2, \mathbf{x}_m) \\ \vdots & \vdots & \ddots & \vdots \\ f(\mathbf{x}_m, \mathbf{x}_1) & f(\mathbf{x}_m, \mathbf{x}_2) & \cdots & f(\mathbf{x}_m, \mathbf{x}_m) \end{bmatrix} \\ &= \begin{bmatrix} c & c & \cdots & c \\ c & c & \cdots & c \\ \vdots & \vdots & \ddots & \vdots \\ c & c & \cdots & c \end{bmatrix} - \begin{bmatrix} h(\mathbf{x}_1, \mathbf{x}_1) & h(\mathbf{x}_1, \mathbf{x}_2) & \cdots & h(\mathbf{x}_1, \mathbf{x}_m) \\ h(\mathbf{x}_2, \mathbf{x}_1) & h(\mathbf{x}_2, \mathbf{x}_2) & \cdots & h(\mathbf{x}_2, \mathbf{x}_m) \\ \vdots & \vdots & \ddots & \vdots \\ h(\mathbf{x}_m, \mathbf{x}_1) & h(\mathbf{x}_m, \mathbf{x}_2) & \cdots & h(\mathbf{x}_m, \mathbf{x}_m) \end{bmatrix} \\ &= c\mathbf{1}\mathbf{1}^T - \mathbf{H}, \end{aligned} \tag{21}$$

where $c = \frac{\pi}{2}$ is a constant, $\mathbf{1} \in \mathbb{R}^m$ is a column vector all 1's. Therefore, it immediately follows

$$\mathbf{z}^T\mathbf{F}\mathbf{z} = c\left(\mathbf{z}^T\mathbf{1}\right)^2 - \mathbf{z}^T\mathbf{H}\mathbf{z} \leq 0, \tag{22}$$

because the first term in (22) is zero whenever $\mathbf{z}^T\mathbf{1} = 0$ (as stipulated for conditionally negative matrices in Definition 1), and because $\mathbf{z}^T\mathbf{H}\mathbf{z} \geq 0$ as $\mathbf{H}$ is the kernel matrix for $h(\mathbf{x}_i, \mathbf{x}_j)$. Hence, the proposed kernel is shown to be positive definite, which is used for classifying time series of manifold points (BoW feature vectors) on a unit sphere $\mathcal{S}^n$.

Abbreviations

ADL: Activities of daily living; BoW: Bag of words; BoW+T: Time-dependent wag of words; CNN: Convolutional neural networks; DTW: Dynamic time warping; ERP: Edit distance with real penalty; FV: Fisher vector; FPR: False positive rate; GA: Global alignment; HOG: Histogram of oriented gradients; LBP: Local binary pattern; LDA: Latent Dirichlet allocation; pLSA: Probabilistic latent semantic analysis; RNN: Recurrent neural networks; REDK: Recursive edit distance kernel; RGB-D: Red green blue + depth; SVM: Support vector machine; STIP: Spatio-temporal interest points; SPD: Symmetric positive definite; SIFT: Scale-invariant feature transform; SPN: Sum-product network; TWED: Time warp edit distance; VLAD: Vector of linearly aggregated descriptors

Acknowledgements

Not applicable.

Funding

Not applicable.

Availability of data and materials

Currently not available.

Authors' contributions

YY is the student author who did the experiments and wrote the manuscript. IG is the supervisor who discussed the proposed method and made the comments on the manuscript draft.

Ethics approval and consent to participate

Not applicable.

Authors' information

Yixiao Yun received a B.Sc. degree in electronic information science and technology from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2009, and M.Sc. degree in communication engineering from Chalmers University of Technology, Gothenburg, Sweden, in 2011. He received the degree of Licentiate of Engineering (Lic.Eng.) and the Ph.D. degree in image and video signal processing from Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden in 2013 and 2016, respectively. His research interests include image processing, visual object tracking, pattern classification, and video activity analysis.

Irene Yu-Hua Gu received the Ph.D. degree in electrical engineering from the Eindhoven University of Technology, Eindhoven, The Netherlands, in 1992. From 1992 to 1996, she was Research Fellow at Philips Research Institute IPO, Eindhoven, The Netherlands, and post dr. at Staffordshire University, Staffordshire, U.K., and Lecturer at the University of Birmingham, Birmingham, U.K. Since 1996, she has been with the Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden, where she has been full Professor since 2004. Her research interests include statistical image and video processing, object tracking and video surveillance, machine learning and deep learning, and signal processing with applications to electric power systems. Dr. Gu was an Associate Editor for the IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, and Part B: Cybernetics from 2000 to 2005, and an Associate Editor with the EURASIP Journal on Advances in Signal Processing from 2005 to 2016. She was the Chair-elect of the IEEE Swedish Signal Processing Chapter from 2002 to 2004. She has been with the Editorial board of the Journal of Ambient Intelligence and Smart Environments since 2011.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Publisher's Note

Received: 13 May 2017 Accepted: 24 October 2017
Published online: 07 November 2017

References

1. Lin, et al., A new network-based algorithm for human activity recognition in videos. IEEE Trans. Circ. Syst. Video Technol. (T-CSVT). **24**(5), 826–841 (2014)
2. I Everts, T JC van Gemert, Gevers, Evaluation of color spatio-temporal interest points for human action recognition. IEEE Trans. Image Process. (T-IP). **23**(4), 1569–1580 (2014)
3. I Laptev, On space-time interest points. Int. J. Comput. Vis. (IJCV). **64**(2/3), 107–123 (2005)
4. G Zhang, M Piccardi, Structural SVM with partial ranking for activity segmentation and classification. IEEE Signal Process. Lett. **22**(12), 2344–2348 (2015)
5. MR Amer, S Todorovic, Sum product networks for activity recognition. IEEE Trans. Pattern. Anal. Mach. Intell. (T-PAMI). **38**(4), 800–813 (2016)
6. H Zhang, LE Parker, CoDe4D: color-depth local spatio-temporal features for human activity recognition from RGB-D videos. IEEE Trans. Circ. Syst. Video Technol. **26**(3), 541–555 (2016)
7. A Karpathy, *et al*, in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. large-scale video classification with convolutional neural networks, (2014), pp. 1725–1732
8. J Donahue, *et al*, in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. long-term recurrent convolutional networks for visual recognition and description, (2015), pp. 2625–2634
9. M Baccouche, *et al*, in *Proceedings of International Workshop on Human Behavior Understanding (HBU)*. sequential deep learning for human activity recognition, (2011), pp. 29–39
10. JM Lee, *Introduction to smooth manifolds*. (Springer, New York, 2012)
11. X Pennec, P Fillard, N Ayache, A Riemannian framework for tensor computing. Int. J. Comput. Vis. (IJCV). **66**(1), 41–66 (2006)
12. V Arsigny, P Fillard, et al., Geometric means in a novel vector space structure on symmetric-positive definite matrices. SIAM. J. Matrix Anal. Appl. (SJMAEL). **29**(1), 328–347 (2008)
13. R Subbarao, P Meer, Nonlinear mean shift over Riemannian manifolds. Int. J. Comput. Vis. (IJCV). **84**(1), 1–20 (2009)
14. A Traumann, et al., Accurate 3D measurement using optical depth information. Electron. Lett. **51**(18), 1420–1422 (2015)
15. O Tuzel, F Porikli, P Meer, in *Proceedings of European Conference on Computer Vision (ECCV)*. Region covariance: a fast descriptor for detection and classification, (2006), pp. 589–600
16. O Tuzel, F Porikli, P Meer, Pedestrian detection via classification on Riemannian manifolds. IEEE Trans. Pattern. Anal. Mach. Intell. (T-PAMI). **30**(10), 1713–1727 (2008)
17. ST Lovett, *Differential geometry of manifolds*, 1st edition. (A K Peters/CRC Press, Natick, 2010)
18. S Jayasumana, *et al*, in *Proceedings of IEEE International Conference on, Digital Image Computing: Techniques and Applications (DICTA)*. Combining multiple manifold-valued descriptors for improved object recognition, (2013), pp. 1–6
19. DG Lowe, Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. (IJCV). **60**(2), 91–110 (2004)
20. N Dadal, B Triggs, Histograms of oriented gradients for human detection. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR). **1**, 886–893 (2005)
21. T Ojala, M Pietikäinen, T Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern. Anal Mach. Intell. (T-PAMI). **24**(7), 971–987 (2002)
22. L Fei-Fei, P Perona, in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. A Bayesian hierarchical model for learning natural scene categories, (2005)
23. PF Felzenszwalb, et al., Object detection with discriminatively trained part based models. IEEE Trans. Pattern. Anal. Mach. Intell. (T-PAMI). **32**(9), 1627–1645 (2010)
24. S Lazebnik, C Schmid, J Ponce, in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, (2006)
25. F Perronnin, J Sánchez, T Mensink, in *Proceedings of European Conference on Computer Vision (ECCV)*. Improving the fisher kernel for large-scale image classification, (2010)
26. H Jégou, M Douze, C Schmid, in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Aggregating local descriptors into a compact image representation, (2010)
27. S Gudmundsson, TP Runarsson, S Sigurdsson, in *Proceedings of IEEE International Joint Conference on Neural Networks (IJCNN)*. Support vector machines and dynamic time warping for time series, (2008)
28. L Chen, R Ng, in *Proceedings of International Conference on Very Large Data Bases (VLDB)*. On the marriage of Lp-norms and edit distance, (2004), pp. 792–803
29. P-F Marteau, Time warp edit distance with stiffness adjustment for time series matching. IEEE Trans. Pattern. Anal. Mach. Intell. (T-PAMI). **31**(2), 306–318 (2008)
30. P-F Marteau, S Gibet, On recursive edit distance kernels with application to time series classification. IEEE Trans. Neural Netw. Learn. Syst. (T-NNLS). **26**(6), 1121–1133 (2015)
31. T Hofmann, B Schölkopf, AJ Smola, Kernel methods in machine learning. Ann. Stat. **36**(3), 1171–1220 (2008)
32. M Cuturi, in *Proceedings of International Conference on Machine Learning (ICML)*. Fast global alignment kernel, (2011)
33. MA Livingston, *et al*, in *Proceedings of IEEE Virtual Reality Workshop (VRW)*. Performance measurements for the Microsoft Kinect skeleton, (2012), pp. 119–120
34. X Chen, A Yuille, in *Proceedings of Advances in Neural Information Processing Systems (NIPS)*. Articulated pose estimation by a graphical model with image dependent pairwise relations, (2014)
35. W Shen, in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Object skeleton extraction in natural images by fusing scale-associated deep side outputs, (2016)
36. G Yu, Z Liu, J Yuan, in *Asian Conference on Computer Vision (ACCV)*. Discriminative orderlet mining for real-time recognition of human-object interaction, (2014)
37. CC Chang, CJ Lin, LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**(3), 27:1–27:27 (2011)
38. J Wang, et al., in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Mining actionlet ensemble for action recognition with depth cameras, (2012)
39. L Xia, JK Aggarwal, in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera, (2013)
40. X Yang, in *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. EigenJoints-based action recognition using Naive-Bayes-Nearest-Neighbor, (2012)

41. M Zanfir, M Leordeanu, C Sminchisescu, in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*. The moving pose: an efficient 3d kinematics descriptor for low-latency action recognition and detection, (2013)
42. IJ Schoenberg, Metric spaces, positivedefinitefunctions. Trans. Am. Math. Soc. (T-AMS). **44**(3), 522–536 (1938)
43. JPR C Berg, P Christensen, *Ressel, Harmonic analysis on semigroups*. (Springer, New York, 1984)
44. P Józiak, Conditionally strictly negative definite kernels. Linear and Multilinear Algebra. **63**(12), 2406–2418 (2015)