Pattern Recognition 86 (2019) 14–26

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Learning principal orientations and residual descriptor for action recognition

Lei Chen[a], Zhanjie Song[b,*], Jiwen Lu[c], Jie Zhou[c]

[a] School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China
[b] School of Mathematics, Tianjin University, Tianjin 300354, China
[c] Department of Automation, Tsinghua University, Beijing 100084, China

ARTICLE INFO

Article history:
Received 13 February 2018
Revised 12 August 2018
Accepted 27 August 2018
Available online 1 September 2018

Keywords:
Action recognition
Unsupervised learning
Trajectories
Principal orientation
Residual value

ABSTRACT

In this paper, we propose an unsupervised representation method to learn principal orientations and residual descriptor (PORD) for action recognition. Our PORD aims to learn the statistic principal orientations and to represent the local features of action videos with residual values. The existing hand-crafted feature based methods require high prior knowledge and lack of the ability to represent the distribution of features of the dataset. Most of the deep learned feature based methods are data adaptive, but they do not consider the projection orientations of features nor the loss of locally aggregated descriptors of the quantization. We propose a method of principal orientations and residual descriptor considering that the principal orientations reflect the distribution of local features in the dataset and the residual of projection contains discriminative information of local features. Moreover, we propose a multi-modality PORD method by reducing the modality gap of the RGB channels and the depth channel at the feature level to make our method applicable to RGB-D action recognition. To evaluate the performance, we conduct experiments on five challenging action datasets: Hollywood2, UCF101, HMDB51, MSRDaily, and MSR-Pair. The results show that our method is competitive with the state-of-the-art methods.

1. Introduction

Recent years have witnessed an increase in various research domains in computer vision [1], such as image processing [2,3], video analysis [4] and multimedia applications [5]. Human activity analysis is an important area in the computer vision and has aroused much attention in several decades. With the development of human activity analysis, action recognition has become a significant research task for its widely used applications. Action recognition has been used in many applications, such as robot interaction, video content comprehension and intelligent surveillance. Recent research on action recognition is interested in realistic datasets which are collected from online videos [6] and movies [7].

The challenge of realistic datasets comes from many aspects, such as variations of views, incomplete observation of human body and illumination variations. In these datasets, the actions in the video have large intra-class variations and the duration of actions also varies greatly. The large intra-class variations observably reduce the ability of hand-crafted features in representing videos and makes these features perform poorly on the datasets. Furthermore, the number of labeled videos is much smaller than that of labeled images in the image classification datasets. The number of labeled data limits the ability of deeply learned features in the learning process. Although deep learning method has attracted great attention mainly due to its success in various visual recognition problems [8], it is still a big challenge in the recognition of realistic videos for human action and it is also far from the human-level performance.

Numerous methods have been proposed for human action recognition. The existing methods based on different types of input data can be divided into three categories: RGB-based action recognition, depth-based action recognition, and skeleton-based action recognition. (1) The methods use the RGB videos as the input data, such as improved dense trajectories (IDT) [9], Two-Stream ConNets [10], and TDD [11]. These methods exploit the evolution of actions with the information of spatial location and temporal continuity. Features are extracted by the predefined strategies with prior knowledge or the deeply learned models [12]. Methods based on the predefined strategies describing actions with abundantly semantic and meaning, but they meet the obstacle that they cannot represent the holistic character of the dataset. Based on the deeply learned models, methods train models with a large number of data, which gives them the good performance especially on the

* Corresponding author.
E-mail addresses: zhanjiesong@tju.edu.cn, 027713@tju.edu.cn (Z. Song).

https://doi.org/10.1016/j.patcog.2018.08.016
0031-3203/

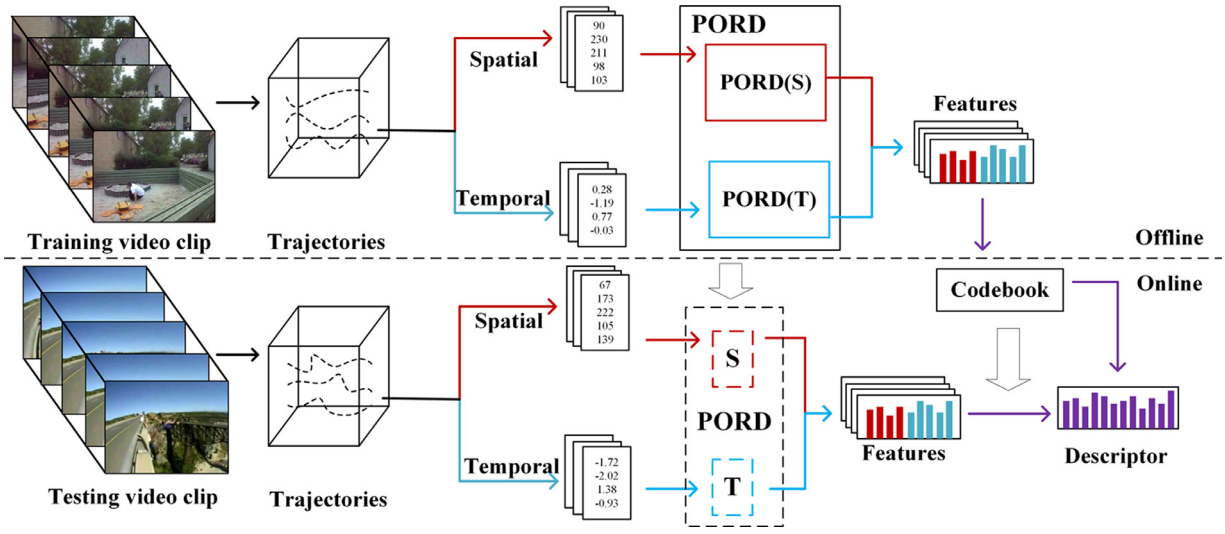

Fig. 1. Pipeline of PORD. The input of training process is raw pixels from video clips based on trajectories. We define the spatial support cells and the temporal support cells to extract the vectors, which are used to learn the mapping matrix of PORD. After getting features from the training set, we use bag-of-feature architecture to generate the codebook. For the testing video clip, we compute the features by the mapping matrix of PORD and generate the descriptor with the codebook.

large-scale dataset. But they have the difficulty of explaining the meaning of each part in models. (2) The methods making use of the depth data exploit the consistency of depth to extract features for action recognition, such as RSDF [13], HON4D [14], and BHIM [15]. These methods using the depth data for mining the consistency of motion to describe actions have two main categories: using only depth data and depth data combing with RGB data. Methods of using only depth data extract features from depth map and represent the human motion by exploiting the consistency in the same region and the divergence in different regions. However, the meticulous degree of depth channel influences the effect of action recognition. Methods combining both the depth data and RGB data simultaneously exploit the motion information of actions in both modalities. (3) The methods using the skeleton information of human body consider the skeleton as features to represent actions, such as [16–18]. These methods use skeleton in two main ways: treating skeletons as features aiming at representing the motion of a human body and mining the geometric architecture from the skeleton to represent actions. Methods using skeletons as features exploit the motion of joints to capture the temporal evolution of actions, which takes the architecture of human body into consideration. Methods extracting the geometric architecture from skeleton aims to describe actions with the pose of the human body, which makes these methods have the advantage of representing different actions. However, the apparent information is lost, which makes these methods have difficulties to describe actions.

In this paper, we design a learning based principal orientations and residual descriptor (PORD) in an unsupervised manner for action recognition. Our method aims to learn a projection weight matrix which consists of several principal orientation vectors and minimizing the loss of quantization. By learning the principal orientations, the projected features can be clustered to the nearest principal orientations, which makes the learned local features more informative. Simultaneously, we exploit the information of residual values of projected features on the non-principal orientations. By maximizing the variance of the residual vectors, the extracted local features are discriminative. Fig. 1 shows the pipeline of our PORD. We extract the trajectories from RGB frames. Then we extract local patches based on the trajectories on RGB and the optical flow field, respectively. The extracted patches are lined into the original vectors with the raw pixel value. After we obtain the vectors both from the RGB frames and optical flow frames, we learn the projection matrix to represent the principal orientations. We use the extracted features to learn a codebook for features embedding. For testing step, we use the projection matrix to extract features and use the learned codebook to encode the testing videos. The final classification step uses the pre-trained classifier to predict the label of testing videos.

The advantages of the proposed descriptors come from: (1) PORD is learned in an unsupervised manner without the constraint of the number of samples in the dataset, which makes PORD adaptive for the dataset of different scales. (2) PORD describes both spatial and temporal information simultaneously and describes the distribution of principal orientations. (3) The loss of quantization in PORD is minimized and the variance of residual vectors are maximized, which makes descriptors informative. Experimental results on three datasets show that our method outperforms existing human action recognition methods and indicate the effectiveness of our method. Moreover, we propose the multi-modality principal orientations and residual descriptor (MPORD) for RGB-D based action recognition. The existing methods for RGB-D dataset do not extract the local information from the depth channel with the cue obtained from the RGB channels. We use trajectories extracted from RGB action videos for jointly learning the projection matrix on RGB channels and the depth channel. This paper is an extended version of our early conference paper [19] and new contributions are as follows:

- We have extended our proposed approach by exploiting both the statistic information from the principal orientations and the discriminative information from the residual vectors by introducing a new objective function. The proposed approach with the new objective function is more general for learning local descriptors of action videos comparing with our previous conference version.
- We have extended our method to deal with RGB-D based action recognition by generalizing the multi-modality objective function which simultaneously learned the principal orientations on RGB channels and the depth channel. The learned features of RGB channels and that of depth channel are complementary, which enhances the representation power of video descriptors.
- We have conducted more experiments in this journal manuscript to evaluate the performance of the proposed descriptor on additional datasets with comparisons to the existing methods for RGB based action recognition. Moreover,

we have also conducted the experiments on the RGB-D based datasets to validate the effectiveness of the extended method for RGB-D based action recognition.

2. Related work

In this section, we briefly review the related works on action recognition. Recent researches on action recognition use different modalities which are captured by different sensors such as RGB camera and the Kinect. We can classify the existing methods into three categories: (1) RGB based action recognition, (2) depth based action recognition, (3) skeleton based action recognition.

2.1. RGB Based action recognition

The RGB camera is the most common device to capture the videos and images. Many works have been proposed for analyzing the features of RGB based videos and images, such as [1–3,5,8,12]. Yu et al. [5] proposed a discriminative coupled dictionary hashing (DCDH) for fast cross-media retrieval. They proposed to learn the coupled dictionary for each modality by considering the side information. Yu et al. [12] developed a novel deep multi-modal distance metric learning (Deep-MDML) method for image ranking. They proposed Deep-MDML to utilize the multi-modal features which contain both click features and visual features to train the ranking model. Zhang et al. [1] proposed an unsupervised deep-learning framework named local deep-feature alignment (LDFA) for dimension reduction. They succeed in modeling both the local and the global characteristics simultaneously and learning a local stacked contractive auto-encoder. Zeng et al. [2] presented a data-driven model coupled with deep auto-encoder (CDA) for single image super-resolution, which aims to learn the intrinsic representation of low-resolution and high-resolution simultaneously and map from low-resolution representation to the corresponding high-resolution representation. Yu et al. [3] proposed a deep multi-task learning algorithm to jointly learn more representative deep convolutional neural networks and more discriminative tree classifier for identifying the privacy-sensitive objects. Yu et al. [8] proposed a generalized multi-modal factorized high-order pooling approach (MFH) aiming at a more effective fusion of multi-modal features. They designed the model by sufficiently exploiting the correlations of multi-modal features and achieved superior VQA performance with DNN architecture. Over the last two decades, most of the works were conducted on RGB videos [20,21]. Based on the features used in the methods, we can divide the RGB based methods into two categories: hand-crafted feature based methods and deep-learned feature based methods.

In the first category, the methods designed the feature extracting rules with prior knowledge. The extracted features reflected the character of the rules and were unconcerned with the samples in the dataset. In [22], Bregonzio et al. proposed an action representation method which differs significantly from the existing interest point based representation in that only the global distribution information of interest points is exploited. In [23], they propose a method which aims at achieving early recognition of ongoing activities. The proposed method is time efficient as it is based on histograms of action poses. In [24], Barnachon et al. proposed a supervised classification method based on a modified sparse model for action recognition by presenting a compound motion and appearance feature is proposed for the interest point at low level. In [25], Liu et al. propose a method for human action recognition based on boosted key-frame selection and correlated pyramidal motion feature representations. In [26], Ijjina et al. propose an approach for human action recognition using genetic algorithms (GA) and deep convolutional neural networks (CNN).

In the second category, the methods used the large of videos to train a deep model for extracting features. The features were data-driven and represented the characteristic of the dataset. Xu et al. [4] proposed a discriminative video representation and trained the model on a large scale video dataset for event detection with limited hardware resources. In [10], Simonyan *et al.* designed an architecture named two-stream ConvNets, which contained spatial and temporal net and explicitly calculated optical flow to capture motion information. Finally, it combined the spatial stream and temporal stream together to predict the label of videos. In [11], Wang et al. utilized deep architectures to learn discriminative convolutional feature maps and constrained by using trajectory pooling to project the convolutional features into discriminative descriptors. Wang et al. [27] discovered the principles to design effective ConvNet architectures for action recognition and proposed a novel temporal segment network (TSN). Tran et al. [28] proposed a simple and effective approach for spatial-temporal features learning using deep 3-dimensional convolutional networks. The method trained on a large scale supervised video dataset and learned features of C3D for action recognition.

2.2. Depth based action recognition

The recent development of depth sensor such as the Kinect has a direct impact on researches of computer vision [29,30]. Depth sensor provides depth information of the scene and human, which can solve problems hard for RGB inputs. Rahmani et al. in [31] proposed a method which used histograms of oriented principle components to align the different views to the same subspace for feature extracting. They achieved higher levels of robustness against viewpoint variations. Lu et al. in [13] proposed binary range-sample descriptors based on τ tests on depth patches. By using the depth feature, they achieved reasonable invariance in scale, viewpoint, and background. Some other researchers used multi-modality analysis by jointly exploiting the information from both RGB videos and depth videos. The main idea was considering that the features from the RGB video and features from the depth video are complementary. Hu et al. [32] proposed a joint learning method by simultaneously exploiting the shared and feature-specific components as the instance of heterogeneous multi-task feature learning. Kong et al. [15] projected RGB and depth data into a learned shared space and used RGB features and depth features jointly by minimizing the rank of their proposed low-rank bilinear classifier.

2.3. Skeleton based action recognition

The skeleton based methods [33] represent actions based on the positions of major body joints. Wang et al. [34] proposed a learning-based maximum margin temporal warping method to align two sequences of actions and match them by using the scores. Vemulapalli et al. [16] proposed a skeleton based representation which explicitly exploited the 3D geometric relationships between different body parts by using rotations and translations in three-dimensional space. Veeriah et al. [17] proposed an architecture for the LSTM neural network by designing a differential gating scheme. Their method enhanced the change of information caused by the salient motion in the adjacent frames. Zhu et al. [18] proposed an architecture of fully connected deep LSTM network and introduced a novel regularization scheme for learning the co-occurrence features from the corresponding skeleton joints. They took the skeleton as the input at each time slot because they considered that the co-occurrences of the joints intrinsically characterized human actions.

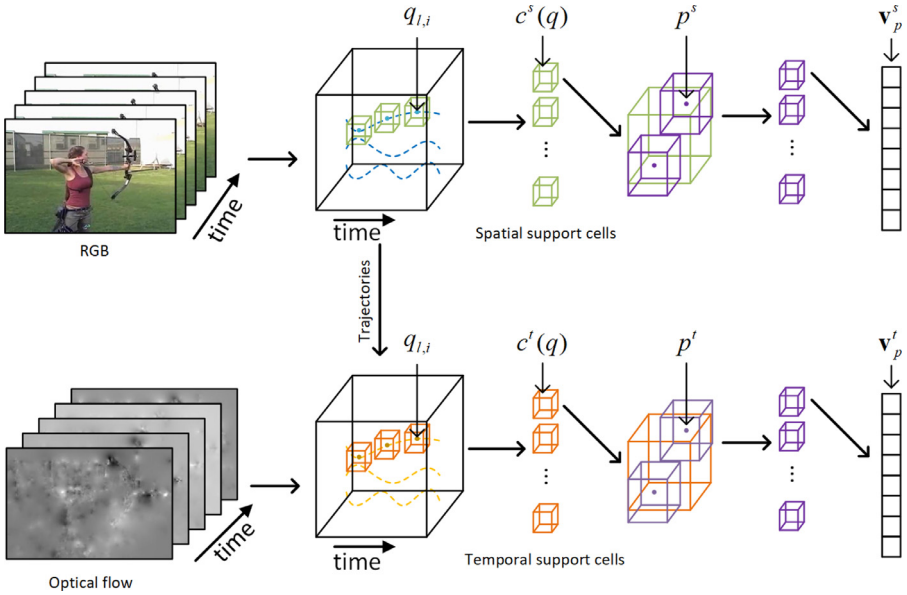

Fig. 2. Spatial and temporal support cell. The top line and the bottom line show the process of extracting original vectors from the spatial support cells and temporal support cells, respectively. The spatial stream and temporal stream share the trajectories extracted from the RGB videos. We generate the spatial support cells by extracting local cuboid around the trajectories. The we densely extract volumes from the spatial support cells and generate the original vectors by stretching the volumes into vectors.

3. Principal orientations and residual descriptor

In this section, we introduce the spatial support cell and the temporal support cell. Then we present our proposed principal orientation and residual descriptor (PORD) based on the RGB data. At last, we give out our optimization method for PORD.

Our feature is learned by the objective function and has the ability to represent the distribution of the principal projection of the dataset. Our PORD is unsupervised and learning based. It inherits the advantages of both the hand-crafted features and the deep-learned features. Our PORD is inspired by the HOG feature to cluster the raw-pixel value vectors into learned clustering centers. The learned clustering centers reflect the principal orientations of local features in the dataset. We use the learned principal orientations to project the features into the subspace and make the residual vectors of the projection discriminative and informative. Moreover, we treat the optical flow field with the similar manner to extract the motion feature of the actions.

3.1. Spatial and temporal support cell

As shown in Fig. 1, PORD is learned from the spatial and the temporal domain individually. We define two corresponding cells for learning: *spatial support cell* and *temporal support cell*. The spatial support cell is extracted from RGB video clips and the temporal support cell is extracted from the corresponding optical flow field. Spatial and temporal support cells are extracted based on trajectories which are obtained by improved dense trajectory (IDT) [9]. We select L trajectories from the training set. Each trajectory ψ_l $(l \in L)$ contains n_f points which are determined by three coordinates (x, y, z). Each point is used as a center to extract a spatial support cell from the RGB video and a temporal support cell from the optical flow field. Fig. 2 represents the details of generation steps of the spatial and temporal cells.

Spatial support cell: Trajectories mainly describe the dense motion of the human body in the video and we extract the spatial support cell from RGB video clips along trajectories. The spatial support cell $c^s(q)$ is represented by the green box in Fig. 2. The spatial support cell $c^s(q)$ has a center at point $q_{l,i}$ $(i \in [1, n_f])$ and radii in three coordinates of the cell are r_x, r_y, r_t. Let $\mathbb{C}^s$ refer to the collection of spatial support cells from training videos and let $\mathbb{P}^s$ denote the collection of all points in all spatial support cells. $p^s \in \mathbb{P}^s$ is an adjacent point close to the $q_{l,i}$ in the extracted spatial support cell, which is used as a center point to extract volumes for original feature extracting. We line all values of pixels in the extracted volume into a vector, which is denoted as $\mathbf{v}_p^s$. $\mathbf{v}_p^s$ is the original feature in the spatial domain, which is used to learn the projection matrix in our principal orientation and residual descriptor.

Temporal support cell: We extract motion features for actions from the optical flow field which is computed from the adjacent frames. Different from the spatial information, temporal information describes the motion of the actions and is more discriminative than the spatial information. The temporal support cell $c^t(q)$ is represented by the yellow box in Fig. 2. The center point of temporal support cell $c^t(q)$ is $q_{l,i}$ $(i \in [1, n_f])$. Let $\mathbb{C}^t$ represent the collection of temporal support cells and $\mathbb{P}^t$ be the collection of all points in all temporal support cells. The vector $\mathbf{v}_p^t$ is the original feature in temporal domain. To simplify the description of our method, we share the description in two domains in the following, since we utilize the similar objective function in both the spatial domain and the temporal domain.

3.2. PORD Feature learning for RGB video

Our proposed PORD is learned in two separate streams for RGB videos. Our PORD aims to learn K projection vectors to extract local features and utilizes the learned features to represent the actions. The learned projection vectors are the principal orientations in the projection subspace for the dataset. Let $w_k \in \mathbb{R}^b$ be the k^{th} projection vector and the projection matrix is denoted as $\mathbf{W} = \{w_1, w_2, \ldots, w_K\}$, where the projection $\mathbf{W}$ is the aim of our learning process. The learned K vectors represent K different orientations in the projection space and determine K principal orientations for the spatial domain and temporal domain. By learning the matrix $\mathbf{W}$, the projected features can represent the local character and the descriptors generated by local features can express the meaning of the actions. The learned matrix maps every vector $\mathbf{v}_p$ into different orientations and quantizes $\mathbf{v}_p$ with the nearest orientations.

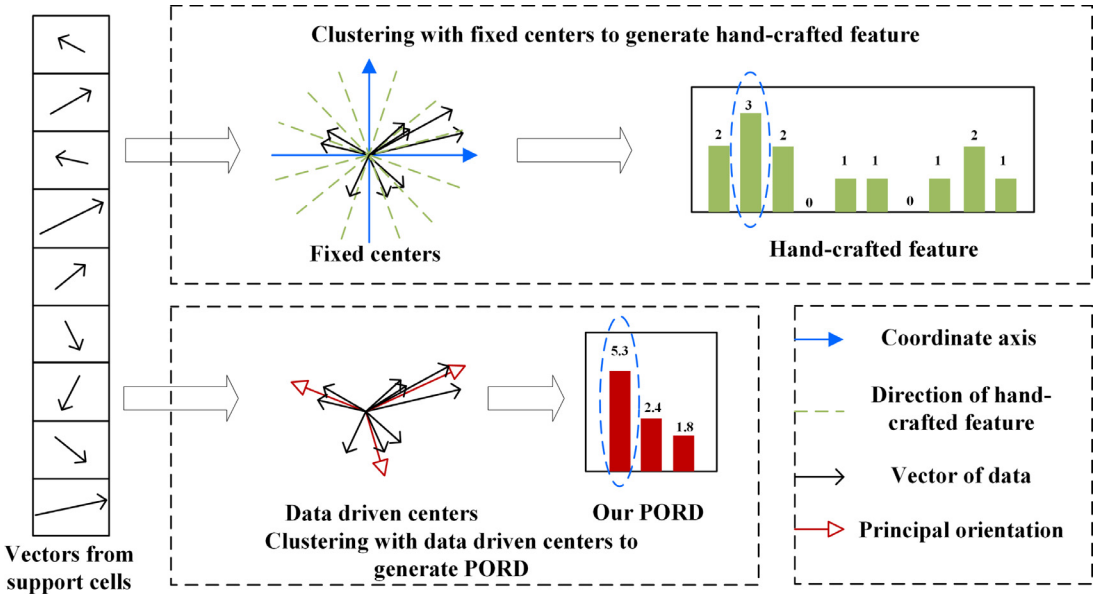

Fig. 3. Comparison between PORD and hand-crafted feature in clustering. Top: Generate hand-crafted feature with fixed clustering centers; Bottom: Generate PORD with data-driven clustering centers.

Fig. 3 shows the aim of our PORD which is to describe the principal orientations of vectors. The fixed clustering centers in hand-crafted features are designed by prior knowledge with the same distance. However, the fixed clustering centers may cluster two similar features into two different centers. Unlike hand-crafted features, our PORD learns the information of distribution of vectors and generates better clustering centers. After the principal orientations are extracted, the residual value vectors are more informative and discriminative. By learning centers, PORD reflects the principal orientations of training data and captures the spatial and temporal features. To distinguish the original feature and the projection feature, we define the projection feature vector as $\mathbf{g}_p$. The elements in the $\mathbf{g}_p$ is defined as $g_p(i)$ and $\mathbf{g}_p$ can be represented by $\mathbf{g}_p = [g_p(1), g_p(2), \ldots, g_p(K)]$. The projection feature vector $\mathbf{g}_p$ is the vector which is mapped from the raw pixels and can be computed as follows:

$$\mathbf{g}_p = \mathbf{W}^{\mathrm{T}}\mathbf{v}_p. \tag{1}$$

The residual vector of $\mathbf{v}_p$ along the non-principal orientations is defined as $\mathbf{r}_p$. The elements in the $\mathbf{r}_p$ can be represented by $\mathbf{r}_p = [r_p(1), r_p(2), \ldots, r_p(K)]$. The residual vector of $\mathbf{r}_p$ is computed as follows:

$$\mathbf{r}_p = \mathbf{g}_p \circ \mathbf{h}_p, \tag{2}$$

where $\mathbf{h}_p$ is to reflect the position of residual values in $\mathbf{g}_p$. The operation of $\circ$ is the element-wise production. The position vector $\mathbf{h}_p$ is computed as follows:

$$\mathbf{h}_p = \Theta(|\mathbf{g}_p|, \|\mathbf{g}_p\|_\infty) \tag{3}$$

where $\|\mathbf{g}_p\|_\infty$ is to represent the position of the maximum value of $|\mathbf{g}_p|$. The function $\Theta(\cdot)$ is a selection function. If the element of $|\mathbf{g}_p|$ is the maximum value which equals $\|\mathbf{g}_p\|_\infty$, the value of corresponding position in $\mathbf{h}_p$ equals to 0, otherwise the value equals 1. $\mathbf{h}_p = [h_p(1), h_p(2), \ldots, h_p(K)]$ denotes the position of the non-maximal value of $|\mathbf{g}_p|$ which corresponds to the position of residual values. We assume that there is only one element which equals to the max value of the $\mathbf{g}_p$ in the vector.

We set three criteria to learn the projection space for making $\mathbf{g}_p$ more informative, more discriminative and easier for clustering:

- The K projection vectors should represent different principal orientations of the training set and be able to describe the distribution of orientations of all volumes centered at the pixels in support cells.
- The variance of residual vector $\mathbf{r}_p$ of $\mathbf{v}_p$ along the non-principal orientations should be maximum, which makes the vectors of projection informative.
- The projection $\mathbf{g}_p$ would preserve the majority of characteristics and natures of $\mathbf{v}_p$ so that the loss of information is minimized during the projection.

Based on the three intentions, we propose our PORD objective function as follows:

$$\begin{aligned}\min_{\mathbf{W}} J(\mathbf{W}) &= J_1(\mathbf{W}) + \lambda_1 J_2(\mathbf{W}) + \lambda_2 J_3(\mathbf{W}) \\ &= -\sum_{p\in\mathbb{P}} \|\mathbf{W}^{\mathrm{T}}\mathbf{v}_p\|_\infty - \lambda_1 \Omega(\mathbf{W}) - \lambda_2 \sum_{c\in\mathbb{C}} \|F(\mathbf{W}, c)\|_2^2 \quad (4)\end{aligned}$$

$$\text{subject to} \quad \mathbf{W}^{\mathrm{T}}\mathbf{W} = I,$$

where $\Omega(\mathbf{W})$ is the variance value of $F(\mathbf{W}, c)$ with all support cells and equals to $\sum_{c\in\mathbb{C}} \|F(\mathbf{W}, c) - \mathbf{u}_F\|_2^2$. $F(\mathbf{W}, c)$ represents the feature of support cell c. $\mathbf{u}_F$ is the mean vector of $F(\mathbf{W}, c)$ among all support cells. $F(\mathbf{W}, c)$ is defined as follows:

$$F(\mathbf{W}, c) = \sum_{p\in c} |\mathbf{W}^{\mathrm{T}}\mathbf{v}_p| \circ \mathbf{h}_p, \tag{5}$$

where $\circ$ operator corresponds to the element-wise product of two vectors.

The different meanings of three terms in (4) for our PORD are as follows: $J_1(\mathbf{W})$ represents the sum of the projection along the principal projection orientation which is corresponding to the max value of $\mathbf{g}_p$. The infinite norm of $\mathbf{g}_p$ preserves the max value along K principal orientations by preserving the energy and orientational nature of $\mathbf{v}_p$. Also, the column vector w_k in $\mathbf{W}$ is a clustering center and $\mathbf{g}_p$ is clustered to the nearest orientation with the weight of infinite norm. When w_k maximizes the sum of the infinite norm of $\mathbf{g}_p$, w_k represents the nearest orientation to $\mathbf{g}_p$ in projection space. It describes one principal orientation and the orientational nature of v_p clearly. For all points in $\mathbb{P}$, when $\mathbf{W}$ minimizes $J_1(\mathbf{W})$ for all projection vectors, $\mathbf{W}$ represents the K principal orientations in the projection space for the training set.

$J_2(\mathbf{W})$ is the sum of variances of $\mathbf{F}(\mathbf{W}, c)$ along non-principal orientations. When $\mathbf{W}$ maximizes the variance of learned clustering

feature codes, the features can be encoded as different as possible. Principal orientations represent the main characteristic of local patches, while residual vectors reflect the details of local patches. Maximizing the variance of the residual vector makes the residual value at every position carry more information. If PORD does not contain residual vectors, the projections along the same principal orientation will be uniformed and lose the power of representing the details of local patches. The learned features which belong to the different principal orientations could easily distinguish the different local patches and the features which are projected into the same principal orientation distinguish the local patches with the projection along the principal orientation. However, when the values of projections along the principal orientation are the same, the residual vector could distinguish the different local patches in details. Maximizing the variance of the residual vectors improves the discriminative power of the learned features.

$J_3(\mathbf{W})$ is the L_1 norm of all residual vectors and it reflects the loss of the projection. By minimizing the third term, we can make the projection on the principal orientation as big as possible. Moreover, we can make the loss of the projection as little as possible.

3.3. Discussion

There are some methods designed for feature embedding, which consider the reconstruction error and the variance of the feature representation by clustering and using the residual values, such as the principal component analysis (PCA) and the vector of aggregate locally descriptor (VLAD).

PCA is a procedure which uses an orthogonal transformation to map possibly correlated variables into a set of linearly uncorrelated variables which are called principal components. For all of the samples, they share these principal components. The principal components in PCA are orthogonal and the importance of components are decreasing. The transformation of PCA is defined in such a way that the first principal component has the largest possible variance. Moreover, PCA aims to minimize the reconstruction error by selecting eigenvectors corresponding to the largest eigenvalues. However, principal orientations in PORD are parallel and learned simultaneously. PORD is to map the raw pixels to different principal orientations, where each of sample belongs to only one principal orientation instead of a set of shared principal components. Different data vectors belong to different principal orientations and have corresponding residual orientations. PORD minimizes the reconstruction error with learning different principal orientations simultaneously and generating the projection of samples to the nearest principal orientations. The learned principal orientations by PORD are orthogonal, which makes both principal orientations and mapped vectors have discriminative information. At the same time, we maximize the variance of residual vector to improve the discriminative power of local patches for representing details.

Commonly-used VLAD generates several clustering centers by k-means whose distribution and the sparsity are dependent by the data. The number of samples belonging to the different centers varies largely. The residual values are computed by the sum of several nearest centers. The representation of features are the lists of residual values. However, PORD learns principal orientations from local patches and represents the distribution of all patches in the training data. The learned orientations have to satisfy the constraint of orthogonality of principal orientations and are limited in a hypersphere, although the principal orientations can be seen as clustering centers. Moreover, the representation of one feature consists of both projection along its principal orientation and the corresponding residual values. The distribution of new mapped features has changed and the sparsity becomes more uniform, which strengthen the representation power of features. The residual values along non-principal orientations are used to discriminate the details of features which have the same principal orientation.

3.4. Optimization for PORD

For computing the optimal $\mathbf{W}$ based on (4), we use the gradient descent method with the curvilinear search algorithm. To make the optimization step easier, we separate the absolute value and the position of maximal value from $\mathbf{W}^\top\mathbf{v}_p$ and rewrite the objective function (4) as follows:

$$\begin{aligned}\min_{\mathbf{W}} J(\mathbf{W}) &= J_1(\mathbf{W}) + \lambda_1 J_2(\mathbf{W}) + \lambda_2 J_3(\mathbf{W}) \\ &= -\sum_{p\in\mathbb{P}} \mathbf{e}\mathbf{W}^\top\mathbf{v}_p - \lambda_1 \sum_{c\in\mathbb{C}} \| \sum_{p\in c} \mathbf{E}^\top\mathbf{W}^\top\mathbf{v}_p \\ &\quad -\frac{1}{N}\sum_{p\in\mathbb{P}} \mathbf{E}^\top\mathbf{W}^\top\mathbf{v}_p\|_2^2 + \lambda_2 \sum_{c\in\mathbb{C}} \| \sum_{p\in c} \mathbf{E}^\top\mathbf{W}^\top\mathbf{v}_p\|_1, \qquad (6)\\ \text{subject to} \quad & \mathbf{W}^\top\mathbf{W} = I,\end{aligned}$$

Where $\mathbf{e}$ is a one-hot vector which contains both the sign and the position of the maximal value of $\mathbf{W}^\top\mathbf{v}_p$. $\mathbf{e}$ and $\mathbf{E}$ are defined as follows:

$$\mathbf{e} = sgn(\mathbf{W}^\top\mathbf{v}_p) \circ \mathbf{z}_p,\ \mathbf{E} = sgn(\mathbf{W}^\top\mathbf{v}_p) \circ (\mathbf{h}_p^\top\mathbf{h}_p), \qquad (7)$$

where the $\mathbf{h}_p$ is defined in (3) and $\mathbf{z}_p$ is defined as:

$$\mathbf{z}_p = \Delta(|\mathbf{W}^\top\mathbf{v}_p|, \|\mathbf{W}^\top\mathbf{v}_p\|_\infty). \qquad (8)$$

where $\Delta(\cdot)$ is a selection function. If the element of $|\mathbf{W}^\top\mathbf{v}_p|$ is the maximum value, the value of the corresponding position in $\mathbf{z}_p$ equals to 1, otherwise, the value equals 0. $\mathbf{z}_p$ has one element which equals 1 and the rest elements all equal 0. The derivative of the objective function $J(\mathbf{W})$ in (6) is as follows:

$$\begin{aligned}\frac{\partial J}{\partial \mathbf{W}} &= \frac{\partial J_1}{\partial \mathbf{W}} + \lambda_1\frac{\partial J_2}{\partial \mathbf{W}} + \lambda_2\frac{\partial J_3}{\partial \mathbf{W}} \\ &= -\sum_{p\in\mathbb{P}} \mathbf{v}_p\mathbf{e}^\top - 2\lambda_1 \sum_{c\in\mathbb{C}} \left(\sum_{p\in c} \mathbf{v}_p\mathbf{\Lambda}^\top\mathbf{E}^\top - \frac{1}{N}\sum_{p\in\mathbb{P}} \mathbf{v}_p\mathbf{\Lambda}^\top\mathbf{E}^\top \right) \\ &\quad +\lambda_2 \sum_{c\in\mathbb{C}} \left(\sum_{p\in c} \mathbf{v}_p\mathbf{\Upsilon}^\top\mathbf{E}^\top \right) \qquad (9)\end{aligned}$$

Where $\mathbf{\Lambda}$ and $\mathbf{Y}$ are computed as:

$$\mathbf{\Lambda} = \sum_{p\in c} \mathbf{E}\mathbf{W}^\top\mathbf{v}_p^\top - \frac{1}{N}\sum_{c\in\mathbb{C}}\sum_{p\in c} \mathbf{E}\mathbf{W}^\top\mathbf{v}_p^\top,\quad \mathbf{\Upsilon} = \sum_{p\in c} \mathbf{E}\mathbf{W}^\top\mathbf{v}_p^\top. \qquad (10)$$

4. Multi-modality principal orientations and residual descriptor D

In this section, we first introduce our proposed multi-modality principal orientation and residual descriptor (MPORD) based on the RGB-D dataset. Then we give out our optimization method for MPORD. Fig. 4 shows the pipeline of MORD.

4.1. MPORD feature learning for RGB-D video

For the RGB-D based action recognition, we propose a method of multi-modality principal orientations and residual descriptor (MPORD). In the method of MPORD, we treat the depth video as the second stream to extract the features. Our MPORD aims to learn K^α projection vectors to map every raw pixel vector $\mathbf{v}_p^\alpha$ from RGB videos and to learn K^β projection vectors to map every raw value vector $\mathbf{v}_p^\beta$, simultaneously. Then we quantize $\mathbf{v}_p^\alpha$ and $\mathbf{v}_p^\beta$ with the nearest projection vectors in the corresponding subspace. Let $w_k^\alpha \in \mathbb{R}^b$ be the k^{th} projection vector for RGB video-based features

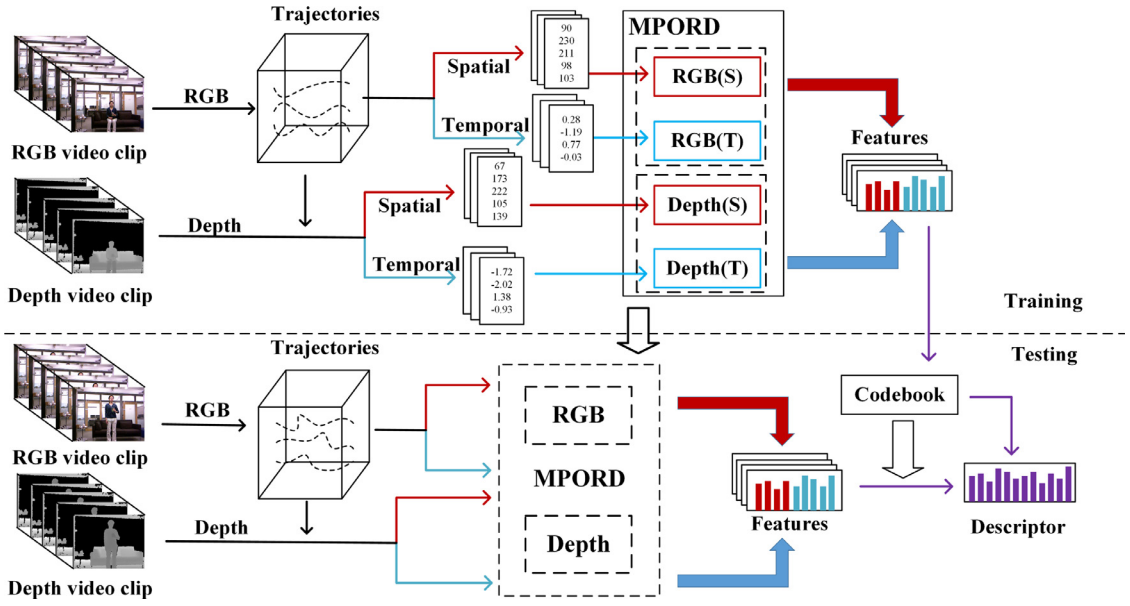

Fig. 4. Pipeline of MPORD. Our MPORD aims to jointly learn two projection weight matrices to generate MPORD for testing video clips. The input of training process is raw pixels from RGB frames and raw values from depth data. We define the spatial and the temporal support cells to extract the vectors similar to that of PORD, which are used to learn the mapping matrix of MPORD. After getting features from the training set, we use bag-of-feature architecture to generate the codebook. For the testing video clip, we compute the features by the mapping matrix of MPORD and generate the descriptor with the codebook.

and let $w_k^\beta \in \mathbb{R}^b$ be the k^{th} projection vector for depth-video based features. Thus the projection matrix for RGB video-based features and for depth video-based features are denoted as follows:

$$\mathbf{W}^\alpha = \{w_1^\alpha, w_2^\alpha, \ldots, w_K^\alpha\},\ \mathbf{W}^\beta = \{w_1^\beta, w_2^\beta, \ldots, w_K^\beta\} \tag{11}$$

The projection matrix of MPORD is a pair of the matrix which is defined as follows:

$$\mathbf{W} \triangleq \begin{bmatrix} \mathbf{W}^\alpha \\ \mathbf{W}^\beta \end{bmatrix} \tag{12}$$

In MPORD, we learn the W^α and W^β jointly and optimize them together. The projection of the RGB input vector $\mathbf{v}_p^\alpha$ is $\mathbf{g}_p^\alpha$ and the projection of the depth input vector $\mathbf{v}_p^\beta$ is $\mathbf{g}_p^\beta$. Base on the PORD, we not only keep the three criteria which are proposed in 3.2, but also we propose the correlation of the two projection matrices. $\mathbf{W}^\alpha$ and $\mathbf{W}^\beta$ are learned from two different modalities but have high relevance. To satisfy the three intentions proposed in 3.2 and the correlation of the two projection matrices, we propose our MPORD objective function as follows:

$$\begin{aligned} \min_{\mathbf{W}^\alpha, \mathbf{W}^\beta} J(\mathbf{W}^\alpha, \mathbf{W}^\beta) = & -\sum_{p\in\mathbb{P}^\alpha} \|(\mathbf{W}^\alpha)^\top \mathbf{v}_p^\alpha\|_\infty - \sum_{p\in\mathbb{P}^\beta} \|(\mathbf{W}^\beta)^\top \mathbf{v}_p^\beta\|_\infty \\ & -\lambda_1 \Omega(\mathbf{W}^\alpha) - \lambda_1 \Omega(\mathbf{W}^\beta) \\ & +\lambda_2 \sum_{c\in\mathbb{C}^\alpha} \|F(\mathbf{W}^\alpha, c)\|_2^2 + \lambda_2 \sum_{c\in\mathbb{C}^\beta} \|F(\mathbf{W}^\beta, c)\|_2^2 \\ & +\lambda_3 CORR(\mathbf{W}^\alpha, \mathbf{W}^\beta) \\ \text{subject to} \quad & (\mathbf{W}^\alpha)^\top \mathbf{W}^\alpha = I,\ (\mathbf{W}^\beta)^\top \mathbf{W}^\beta = I, \end{aligned} \tag{13}$$

where $F(\mathbf{W}^\alpha, c)$ represents the feature of support cell c from the RGB video and $F(\mathbf{W}^\beta, c)$ represents the feature of support cell c from the depth data. $\Omega(\mathbf{W}^\alpha)$ is the variance value of $F(\mathbf{W}^\alpha, c)$ with all support cells and $\Omega(\mathbf{W}^\beta)$ is the variance value of $F(\mathbf{W}^\beta, c)$. $\mathbf{u}_F^\alpha$ is the mean vector of $F(\mathbf{W}^\alpha, c)$ among all RGB based support cells and $\mathbf{u}_F^\beta$ is the mean vector of $F(\mathbf{W}^\beta, c)$ among all depth based support cells. $F(\mathbf{W}^\alpha, c)$ and $F(\mathbf{W}^\beta, c)$ are defined as follows:

$$\begin{aligned} F(\mathbf{W}^\alpha, c) &= \sum_{p\in c} |(\mathbf{W}^\alpha)^\top \mathbf{v}_p^\alpha| \circ \mathbf{h}_p^\alpha, F(\mathbf{W}^\beta, c) \\ &= \sum_{p\in c} |(\mathbf{W}^\beta)^\top \mathbf{v}_p^\beta| \circ \mathbf{h}_p^\beta, \end{aligned} \tag{14}$$

where $\mathbf{h}_p^\alpha$ and $\mathbf{h}_p^\beta$ are the vectors to mark the positions where the corresponding projection orientation is non-principal. These two vectors are computed as follows:

$$\begin{aligned} \mathbf{h}_p^\alpha &= \Theta(|(\mathbf{W}^\alpha)^\top \mathbf{v}_p^\alpha|, \|(\mathbf{W}^\alpha)^\top \mathbf{v}_p^\alpha\|_\infty), \\ \mathbf{h}_p^\beta &= \Theta(|(\mathbf{W}^\beta)^\top \mathbf{v}_p^\beta|, \|(\mathbf{W}^\beta)^\top \mathbf{v}_p^\beta\|_\infty), \end{aligned} \tag{15}$$

where the function $\Theta(\cdot)$ is defined similarly to that of (3). $\mathbf{h}_p^\alpha$ and $\mathbf{h}_p^\beta$ are the vectors which denote the position of the non-maximal value of $|(\mathbf{W}^\alpha)^\top \mathbf{v}_p^\alpha|$ and $|(\mathbf{W}^\beta)^\top \mathbf{v}_p^\beta|$. We assume that there is only one element which equals to the max value of the $(\mathbf{W}^\alpha)^\top \mathbf{v}_p^\alpha$ and the $(\mathbf{W}^\beta)^\top \mathbf{v}_p^\beta$. In (13), $CORR(\mathbf{W}^\alpha, \mathbf{W}^\beta)$ can be represented as $tr(\mathbf{W}^\top \Phi \mathbf{W})$ and $tr(\cdot)$ refers to the operation of trace. The Φ is computed as follows:

$$\Phi = \begin{bmatrix} \mathbf{0} & \mathbf{W}^\alpha \\ \mathbf{W}^\beta & \mathbf{0} \end{bmatrix} \tag{16}$$

5. Experiments

5.1. Datasets

We evaluated PORD on RGB based action recognition dataset: Hollywood2 [7], UCF101 [35], and HMDB51 [6]. Foe MPORD, we evaluated the method on RGB-D based action recognition dataset: MSR-Pair [36] and MSRDaily [14]. Fig. 5 shows examples of these datasets.

Hollywood2: The Hollywood2 dataset is gathered from 69 different Hollywood movies and contains 12 classes of human actions, which has 1707 video clips. Based on the dataset setup in [7], 823

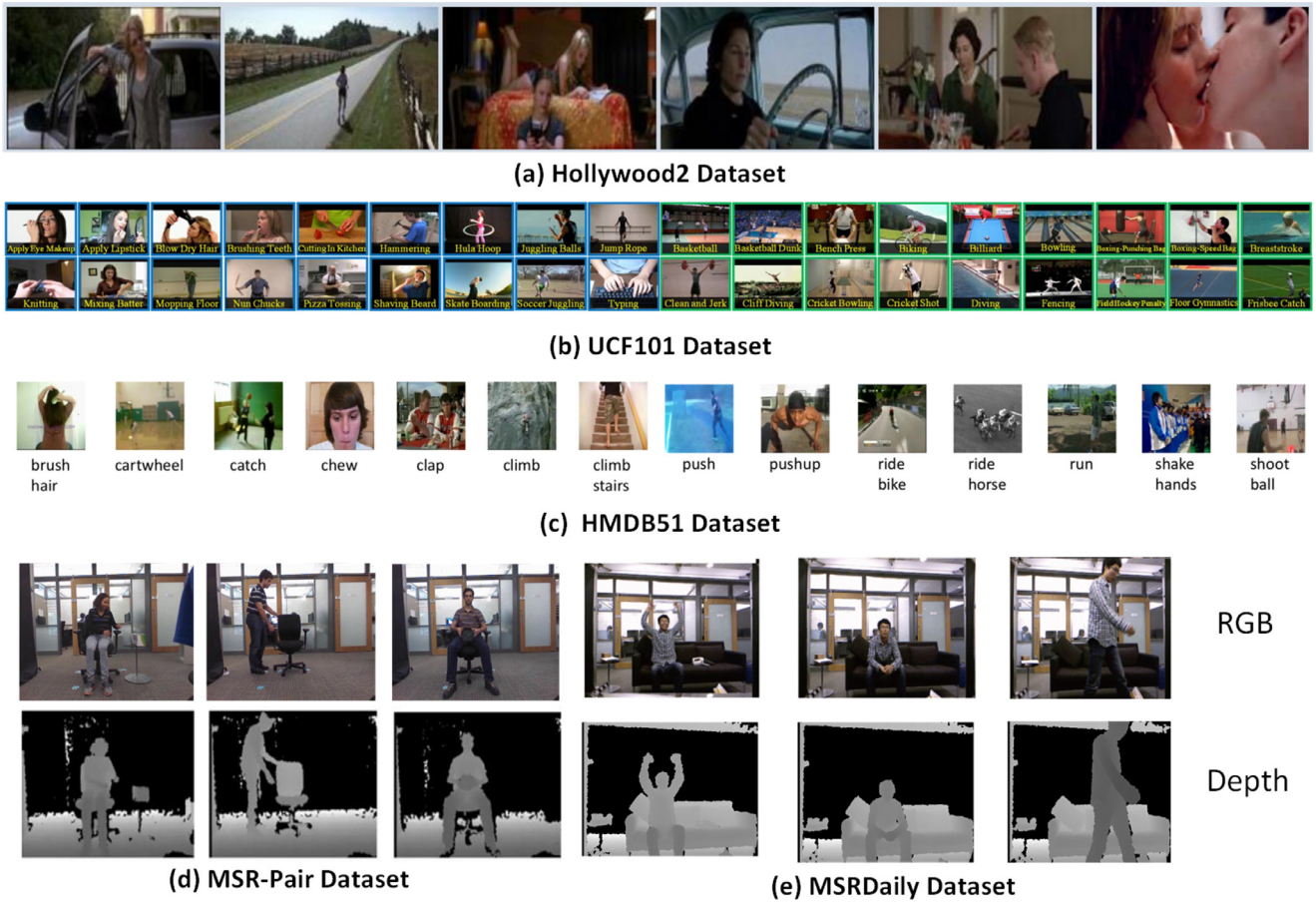

Fig. 5. Examples of the RGB video based datasets. (a), (b), and (c) show the examples of Hollywood2, UCF101, and HMDB51 dataset, respectively, which are RGB based datasets. (d) shows the examples of MSR-Pair dataset and (e) shows the examples of MSRDaily dataset, which are RGB-D based datasets. The first line shows the RGB frames in videos and the second line shows the corresponding depth map.

video clips are used for training and the rest 884 video clips for testing. The performance on the dataset is measured by mean average precision (mAP) over all classes.

UCF101: UCF101 action recognition dataset has 101 categories and is collected from the YouTube, which are realistic action videos. There are 13,320 videos which are divided into 25 groups. There are three splits proposed by [35], the results reported in the experiments are the averaged results of three splits.

HMDB51: The HMDB51 dataset contains 51 classes and is collected from a large amount of YouTube videos, which contains 6766 video clips. [6] proposed three splits for cross-validation and the measurement of performance is average accuracy over all three splits. For every class in one split, 70 video clips are selected for training and 30 video clips are selected for testing.

MSR-Pair: The MSR-Pair is a 3D dataset for RGB-D action recognition. The actions are collected in pairs, which the two actions of one pair have similar trajectories and similar objects. This dataset provides 6 pairs of action classes, which is captured from 10 subjects, each one 3 times. The first five subjects are kept for testing and the remains are for training. The total number of RGB-D videos in this dataset is 360 .

MSRDaily: The MSRDaily dataset is captured by the Kinect device. There are 16 categories of actions in the dataset, which has 10 subjects. For every subject, the same action is captured twice, once for standing position, and once for sitting position. There are $16 \times 10 \times 2 = 320$ files for each channel.

5.2. Implementation details

In our experiment of PORD and MPORD, there were three steps: learning projection matrix, representing every video clip with PORD and classifying on testing set. **Learning step:** For each dataset, we used 15 frames as the length of the trajectories and extracted support cells at all points along trajectories. In the temporal domain, we extracted cells from horizontal and vertical components, respectively. For every cell, we set the radii of the cell in three coordinates as $r_x = r_y = r_t = 7$ in learning step for the computation complexity. For every point in support cell, we fixed the width of volumes as 3 pixels. The value of λ_1 was 1 and λ_2 was 0.1 during the training step. For MPORD, λ_3 was 0.1. **Representation step:** We used the projection matrix learned in the first step to represent video clips. To generate codebook, the number of features for k-means was fixed to 10^5 and the number of clustering centers was 4000, which was the same as that in [9]. Then we used the codebook to compute the descriptor of every video clip. **Classification step:** We trained a χ^2 kernel with training set and used the non-linear SVM as the classifier to predict the label of the testing videos.

5.3. Comparison with existing methods based on RGB videos

Results on Hollywood2: Table 1 shows the results of our PORD compared with several existing methods on Hollywood2 dataset. PORD achieves the state-of-the-art performance in the compari-

Table 1
Comparison of recognition accuracy with existing methods on Hollywood2 (%).

Method	H/L	Hollywood2
Vig et al. [37]	Hand-crafted	59.4
Jiang et al. [38]	Hand-crafted	59.5
Mathe et al. [39]	Hand-crafted	61.0
Jain et al. [40]	Hand-crafted	62.5
Wang et al. [9]	Hand-crafted	63.0
PORD(S+T)	Learned	69.3
PORD(S+T) + hand-crafted [9]	Both	70.2

Table 2
Comparison of recognition accuracy with the state-of-the-arts on UCF101 and HMDB51 (%).

Method	S/U	UCF101	HMDB51	Year
Wang et al. [9]	Unsupervised	80.7	52.1	2013
Cai et al. [44]	Unsupervised	83.5	55.9	2014
Srivastava et al. [42]	Unsupervised	75.8	44.0	2015
Peng et al. [41]	Unsupervised	87.9	61.1	2016
Liu et al. [45]	Unsupervised	76.3	51.4	2017
Zisserman et al. [10]	Supervised	88.0	59.4	2014
Lin et al. [46]	Supervised	88.1	59.1	2015
Wang et al. [11]	Supervised	91.5	65.9	2015
Yang et al. [47]	Supervised	91.6	61.8	2016
Carreira et al. [43]	Supervised	93.4	66.4	2017
PORD(S+T)	Unsupervised	89.1	58.5	-
PORD(S+T) + deep-learned [10]	Supervised	95.1	67.3	-

son with hand-crafted methods. Compared with the IDT, PORD achieved an improvement of 6.3%, which demonstrated that by learning the principal orientations and constraining residual value, our PORD represented the action videos better. After combining the hand-crafted features from [9], there was a little improvement. The reason was that our PORD had learned the local discriminative information which was contained in [9].

Results on UCF101: Table 2 shows the results of our PORD compared with the state-of-the-art methods based on both supervised and unsupervised learning on UCF101 and HMDB51. We analyzed the results on two datasets respectively. On UCF101, PORD achieves the state-of-the-art performance comparing with unsupervised methods and performs well comparing with the supervised methods. By simply combining one supervised feature, PORD also achieves the state-of-the-art performance in the supervised manners. PORD improved 8.4% comparing with [41], which came from the learned distribution of local features in the whole dataset. Compared with [42], our result improves over 10.0% without combining the other feature. The improvement was caused by utilizing the trajectories as the cue of actions in the learning process. By combining the feature from [10] whose performance was much worse than that of [43], PORD outperformed [43] with an improvement of 1.7%.

Results on HMDB51: Table 2 shows the results of PORD compared with several state-of-the-art methods on HMDB51. The dataset of HMDB51 is much more challenge than UCF101 for the reason that the environment of HMDB51 is more complex and the variance of the same action is larger. The performance on HMDB51 is far from that of UCF101. In contrast of [42] which was unsupervised deep learning method, our PORD outperformed 14.5% by exploiting the statistically local character of action videos. After combining the deep feature from Zisserman et al. [10] which was the basic deep learning method for action recognition, PORD outperformed the state-of-the-art with about 1.0%. The results of TDD improved 4.7% on HMDB51 than PORD. Because the method of TDD extracted hierarchically global features with spatial-temporal networks proposed by Two-Stream [10], which extracted global features in a supervised manner. However, PORD learned the principal orientations from local features in an unsupervised manner. Moreover, PORD achieved better results by combining with the Two-Stream feature. The improvement demonstrated that PORD had learned the local character which had not been extracted in the deep features.

Table 3
The recognition accuracy of cross-validation for PORD on three datasets (%).

Dataset	Descriptor	Experiment			Averaged
		Split 1	Split 2	Split 3	
Hollywood2	PORD(S)	59.9	58.8	59.5	59.4 ± 0.56
	PORD(T)	64.1	65.3	64.2	64.5 ± 0.67
	PORD(S+T)	69.3	68.8	69.1	69.1 ± 0.25
UCF101	PORD(S)	74.9	75.9	75.7	75.5 ± 0.53
	PORD(T)	79.7	81.0	80.2	80.3 ± 0.66
	PORD(S+T)	88.4	89.3	89.6	89.1 ± 0.62
HMDB51	PORD(S)	43.7	44.3	44.0	44.0 ± 0.30
	PORD(T)	51.9	53.0	53.5	52.8 ± 0.82
	PORD(S+T)	58.4	58.7	58.4	58.5 ± 0.17

Table 4
The recognition accuracy of PORD on UCF101 and HMDB51 datasets (%).

Descriptor	UCF101				HMDB51			
	w/o	fixed	CNN	PORD	w/o	fixed	CNN	PORD
PORD(S)	72.1	69.8	65.6	75.5	39.2	38.5	32.1	44.0
PORD(T)	76.9	76.5	70.2	80.3	48.8	47.3	39.0	52.8
PORD(S+T)	85.6	83.2	77.9	89.1	55.0	53.4	45.8	58.5

Table 3 shows the detailed results of three experiments on three datasets. For Hollywood2 dataset, we conducted the experiment with the training set and testing set designed by [7], in which the training set and the testing set are fixed. For fair comparisons, we used the same setting for three experiments to obtain the statistics of the results. The 'Averaged' refers to the statistics of the three experiments and contains the averaged accuracy with the standard deviation. The standard deviations of the three experiments on spatial steam and temporal stream are 0.56% and 0.67%, respectively. The standard deviation reduce to 0.25% with feature combination of two streams. For UCF101 dataset, we conducted the experiments and obtained the results on three splits proposed by [35]. The largest gap of results in the three splits is 1.3% of separate stream and 1.2% of combination with two streams. The largest gap reduces to 0.7% by combining IDT features and to 0.6% by combining Two-Stream features. From the results, the combination of different features improves the robustness of representing actions. For HMDB51 dataset, we followed the settings proposed by [6] which divide the dataset into three different splits for cross-validation. The largest difference of results on three splits obtains 1.6% between split 1 and split 3 on the temporal stream, which illustrates that the averaged results could reflect the effectiveness of our methods.

To demonstrate the effectiveness of our method, we conducted some ablation studies about without maximizing the variance of residual vectors, using fixed orientations, and extracting CNN features as the original visual features. In Table 4, 'w/o' refers to the results of without using the second term in our objective function and PORD uses maximizing the variance of residual vectors, respectively. From the comparison between two settings, the performances of PORD are higher than those of 'w/o'. On UCF101 dataset, the accuracy of recognition increases 3.4% on spatial stream and temporal steam, respectively, and improves at least 3.5% on the combination of two streams. The experiments on the other two datasets achieve the consistent results.

In Table 4, The 'fixed' refers to that the orientations are fixed and orthonormal. The subsequent steps of generating codebook and classification are the same as those of PORD. The performances of PORD improve at least 5.5% on spatial stream and 3.8% on temporal stream. For combination of two streams, the performance in-

Table 5
Comparison of recognition accuracy with the state-of-the-arts on MSRDaily and MSR-Pair (%).

Method	category	MSRDaily	MSR-Pair	Year
Oreifej et al. [36]	hand-crafted	80.0	96.67	2013
Wang et al. [34]	shallow+supervised	88.8	97.22	2013
Wang et al. [49]	shallow+unsupervised	85.8	82.22	2014
Lu et al. [13]	shallow+unsupervised	95.6	-	2014
Hu et al. [32]	shallow+unsupervised	95.0	-	2015
Rahmani et al. [31]	hand-crafted	88.8	98.3	2016
Shahroudy et al. [50]	shallow+unsupervised	91.3	100.0	2016
Shahroudy et al. [48]	deep+unsupervised	97.5	100.0	2017
MPORD(RGB)	shallow+unsupervised	89.4	100.0	-
MPORD(Depth)	shallow+unsupervised	98.8	98.3	-
MPORD(RGB-D)	shallow+unsupervised	99.4	100.0	-

creases 6.9% and 5.1% on UCF101 and HMDB51, respectively. The results show that PORD obtained an improved performance with learned orientations.

In Table 4, 'CNN' refers to using CNN features as the original features to learn principal orientations. We used the local patches whose centers were joints of trajectories and extracted the CNN features with the pre-trained VGG-16 network on the local patches. The descriptor of one trajectory was the averaged vector of all features extracted along the trajectory. Then the dimension of the extracted descriptors was reduced to 256 and the generated features were used for learning the principal orientations. The results of PROD are higher than that of CNN feature for three RGB based datasets. The results of PROD improve at least 9.9% on spatial stream and 10.1% on temporal stream, since the CNN features are sparse and hard to learn the principal orientations. For the combination of two streams, the improvement reaches 11.2% on UCF101 and 12.7% on HMDB51, respectively. The improvement demonstrates that the raw-pixel is more effective than CNN for learning the local features.

5.4. Comparison with existing methods based on RGB-D videos

Results on MSRDaily: Table 5 shows the results of MPORD compared with several state-of-the-art methods for RGB-D based action recognition. Compared with the hand-crafted features based methods, MPORD improved at least 10%, which get benefit from the learned distribution of local features on both RGB channels and depth channel. The highest accuracy of shallow learning methods performed 95.6%. MPORD still achieved an improvement of 3.8%, which came from the joint learning. The state-of-the-art result for MSRDaily dataset was obtained by Shahroudy et al. [48], which used the unsupervised deep learning. MPORD reached the accuracy of 99.4%, which surpassed [48] with 1.9%. The reason of improvement was that MPORD not only considered the joint learning for two modalities, but also sufficiently exploited the distribution of local features on spatial and temporal domains.

Results on MSR-Pair: The MSR-Pair dataset is challenging for the similarity pair-base actions. From the results in Table 5, we observed that the hand-crafted based methods obtained the high accuracy. But there were still several videos with wrongly predicting labels. The reason was that the pre-designed extracting rules had the ambiguity on two similar actions with different labels. However, MPORD was learned from the training set and was adaptive for the dataset. MPORD also achieved the 100.0% accuracy and both channels provided the discriminative information.

We showed the cross-validation results and the statistic results in Table 6 for RGB-D based action recognition. For MSRDaily dataset, we conducted the experiments by using the first 5 subjects for testing and the rest for training, which followed the settings of [14] and represented with #1. For cross-validation, we random selected 5 subjects for training and used the rest for testing, which are represented as #2 and #3. For three different settings of channels, the standard deviation of temporal stream is smaller than that of spatial stream and the combination of two streams gets the smallest standard deviation. The results of #2 achieve better performance than #1 and the increases of results vary from 0.1% to 1.0%. The #3 obtains comparative results with #1 and the differences of results vary from −0.6% to 1.2%. The results demonstrate that the original setting is harder than the setting of random selection, but statistic results are consistent with that of the original setting. For MSR-Pair dataset, #1 refers to the settings proposed by [36] which used the last 5 actors for training and the rest for testing. For cross-validation, we conducted two experiments with randomly selecting 5 actors for training and the rest for testing, which are represented by #2 and #3 . The results on MSR-Pair show that random selection makes a little difference of the results. For fair comparisons, we used the results of #1 comparing with other methods.

5.5. Evaluation of MPORD

Table 7 presents the results of the different experimental strategies: RGB channels, depth channel and RGB-D channels. In every strategy, we compared the spatial stream, temporal stream and both streams.

RGB channels: The performance of MPORD on two datasets had a gap over 10% of accuracy, which was caused by the large variation of actions and the complexity of background on MSRDaily. The temporal domain achieved a better performance over 10% compared with using the spatial domain. With the combination of two streams, the accuracy obtains a better performance which demonstrates that the features extracted from the spatial and the temporal domains are complementary.

Depth channel: The accuracy of depth channel on two datasets both surpassed 93%, which proved that the features extracted by MPORD from the depth channel was more robust than those of RGB channels. On MSRDaily, the accuracies of RGB channels is lower than those of MSR-Pair, which was caused by the complex environment and actions. However, the features extracted by MPORD on depth channel were robust for the environment variation. Because, the depth channel mainly described the information of distance and motion with little appearance information.

RGB-D channels: MPORD achieved the state-of-the-art performance on both datasets. The performances of action recognition achieve 99.4% on MSRDaily and 100.0% on MSR-Pair. The strategy of using two channels and two streams obtained the best performance proved that the features from two modalities described the action from the different aspects and the features from two streams captured the complementary information for representing the actions.

Table 6
The recognition accuracy of cross-validation for MPORD on MSRDaily dataset (%).

Descriptor	MSRDaily				MSR-Pair			
	#1	#2	#3	Averaged	#1	#2	#3	Averaged
RGB(S)	81.3	82.3	80.7	81.4 ± 0.81	95.0	97.2	96.7	96.3 ± 1.15
RGB(T)	88.1	89.1	89.3	88.8 ± 0.64	100.0	99.4	100.0	99.8 ± 0.35
RGB(S+T)	89.4	89.7	89.4	89.5 ± 0.17	100.0	100.0	100.0	100.0 ± 0.00
Depth(S)	97.5	97.9	97.8	97.7 ± 0.21	93.9	94.4	95.0	94.4 ± 0.55
Depth(T)	98.8	98.9	99.1	98.9 ± 0.15	98.9	98.6	97.5	98.3 ± 0.74
Depth(S+T)	98.8	99.1	99.0	99.0 ± 0.15	98.3	98.3	98.9	98.5 ± 0.35
RGB-D(S)	95.6	95.8	96.8	96.1 ± 0.64	100.0	100.0	100.0	100.0 ± 0.00
RGB-D(T)	98.1	98.4	98.9	98.5 ± 0.40	100.0	100.0	100.0	100.0 ± 0.00
RGB-D(S+T)	99.4	99.7	99.6	99.6 ± 0.15	100.0	100.0	100.0	100.0 ± 0.00

Table 7
The recognition accuracy of MPORD on MSRDaily and MSR-Pair datasets (%).

Descriptor	MSRDaily				MSR-Pair			
	w/o	fixed	CNN	MPORD	w/o	fixed	CNN	MPORD
RGB(S)	75.7	76.5	60.3	81.3	89.7	91.7	73.3	95.0
RGB(T)	80.8	81.2	67.1	88.1	90.1	93.4	77.4	100.0
RGB(S+T)	85.6	85.9	70.5	89.4	92.0	95.0	80.0	100.0
Depth(S)	90.0	90.1	66.5	97.5	88.8	90.2	75.3	93.9
Depth(T)	91.7	89.7	69.7	98.8	89.9	93.1	77.1	98.9
Depth(S+T)	93.0	91.3	71.2	98.8	91.0	94.7	79.9	98.3
RGB-D(S)	88.9	88.4	65.7	95.6	93.4	92.1	77.7	100.0
RGB-D(T)	91.7	90.6	72.2	98.1	94.7	94.2	79.1	100.0
RGB-D(S+T)	94.4	93.3	78.9	99.4	95.0	96.0	83.5	100.0

Table 8
The time cost (hours) on different datasets.

Time	PORD			MPORD	
	Hollywood2	UCF101	HMDB51	MSRDaily	MSR-Pair
Training Time(h)	10	25	30	3	2
Testing Time(h)	0.5	1.5	2.0	0.5	0.5

For the ablation study for MPORD, we conducted the experiments to compare MPORD with the settings of without maximizing the variance of residual vectors, using fixed orientations, and extracting CNN features as the original visual features. Table 7 shows the results of comparison on MSRDaily and MSR-Pair datasets. In Table 7, 'w/o' refers to learning orientations without maximizing the variance of residual vectors. With maximizing the variance of residual vectors, the representation power of learned features has increased. The accuracy of MPORD has improved at least 2.5% on the RGB based datasets and 3.0% on the RGB-D based datasets than 'w/o', which demonstrates the effectiveness of maximizing the variance of residual vectors.

The 'fixed' in Table 7 refers to using the fixed orientations to extract features. The results of PORD which learn orientations increase at least 3.3% than the fixed orientations since the learned features are more discriminative by representing the distribution of the data. For RGB channels, the results of learned orientations improve at least 3.3% on the separate stream and 3.5% on the combination of two streams. For depth channel, the learned orientations achieve better performances with an increase of 9.1% on the temporal stream and improve at least 3.6% on the combination of two streams. The results of joint learning obtain an improvement of 6.1% on MSRDaily dataset and 4.0% on MSR-Pair with both streams. Comparing the results of RGB channels and depth channel, depth channel achieves better performance since the learned features from depth channel capture more exact information with less noise from the background.

The 'CNN' in Table 7 refers to using the CNN features as the original features for learning. For RGB channels, the settings are the same as those of Table 7. For depth channel, we extracted features from depth channel with the pre-trained VGG-16 network by using only one channel of depth data. Since the pre-trained model is not for extracting features on depth map, the performance of depth channel decreases a little on MSR-Pair dataset. The results of PORD on RGB channels and depth channel improve 20.5% and 16.5% on MSRDaily and MSR-Pair by using raw-pixel. The raw-pixel is effective for MPORD to learn the projection matrix from local patches on both RGB channels and depth channel.

We analyze computational complexity from two aspects: the complexity of computing the objective function and the complexity of computing the deviation. From the equation (4), the computational complexity of the objective function is $n_P \times O(DK)$, where D and K are the sizes of projection matrix $\mathbf{W}$ and n_P is the number of all points in the spatial support cells or the temporal support cells. We assume that the average of features, $\mathbf{u}_F$, is computed for one time in one epoch. From the equation (14) in our revised manuscript, the computational complexity of the three terms in the deviation is $n_P \times O(DK)$. In our experiment, we set the max iteration as n_I. The complexity of the whole training process is $n_I \times n_P \times O(DK)$. We conducted experiments to compare the time cost of our method on different datasets with 12 CPUs of E5-2698 v4 @ 2.20GHz. Table 8 shows the results of time cost for training process and testing process, respectively. From the results, we can find out that the time cost keeps positive correlation to the number of samples.

6. Conclusions

This paper proposes the principal orientations and residual descriptor (PORD) to describe human action video clips. PORD ex-

ploits the spatial and temporal information in the local cuboid and learns features with an unsupervised architecture from raw pixels of video clips. PORD learns the distribution of principal orientations of dataset and shares the merits of the spatial and the temporal information. The learning process does not require any additional training data which makes PORD have a good performance on the small dataset. We show that the performance is significantly improved comparing with hand-crafted methods and is competitive with deep learning methods on three public popular datasets. Moreover, we extend PORD to MPORD for RGB-D based action recognition by jointly learning the features of RGB videos and the those of depth videos. The correlation and the consistency of RGB and depth channels improves the representation of the learned features for RGB-D based videos. The experimental results on RGB-D dataset demonstrate the effectiveness of MPORD.

Acknowledgments

This work was supported in part by the National Key Research and Development Program of Ch ina under Grant 2017YFA0700802, in part by the National Natural Science Foundation of China under Grant 61672306, Grant U1713214, Grant 61572271, Grant 91746107, in part by the Shenzhen Fundamental Research Fund (Subject Arrangement) under Grant JCYJ20170412170602564.

References

[1] J. Zhang, J. Yu, D. Tao, Local deep-feature alignment for unsupervised dimension reduction, TIP 27 (5) (2018) 2420–2432.
[2] K. Zeng, J. Yu, R. Wang, C. Li, D. Tao, Coupled deep autoencoder for single image super-resolution, IEEE Trans. Cybern. 47 (1) (2017) 27–37.
[3] J. Yu, B. Zhang, Z. Kuang, D. Lin, J. Fan, iprivacy: image privacy protection by identifying sensitive objects via deep multi-task learning, TIFS 12 (5) (2017) 1005–1016.
[4] Z. Xu, Y. Yang, A.G. Hauptmann, A discriminative CNN video representation for event detection, in: CVPR, 2015, pp. 1798–1807.
[5] Z. Yu, F. Wu, Y. Yang, Q. Tian, J. Luo, Y. Zhuang, Discriminative coupled dictionary hashing for fast cross-media retrieval, in: ACM SIGIR, 2014, pp. 395–404.
[6] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, T. Serre, HMDB: a large video database for human motion recognition, in: ICCV, 2011, pp. 2556–2563.
[7] M. Marszalek, I. Laptev, C. Schmid, Actions in context, in: CVPR, 2009, pp. 2929–2936.
[8] Z. Yu, J. Yu, C. Xiang, J. Fan, D. Tao, Beyond bilinear: generalized multimodal factorized high-order pooling for visual question answering, TNNLS (99) (2018) 1–13.
[9] H. Wang, C. Schmid, Action recognition with improved trajectories, in: ICCV, 2013, pp. 3551–3558.
[10] K. Simonyan, A. Zisserman, Two-stream convolutional networks for action recognition in videos, in: NIPS, 2014, pp. 568–576.
[11] L. Wang, Y. Qiao, X. Tang, Action recognition with trajectory-pooled deep-convolutional descriptors, in: CVPR, 2015, pp. 4305–4314.
[12] J. Yu, X. Yang, F. Gao, D. Tao, Deep multimodal distance metric learning using click constraints for image ranking, IEEE Trans. Cybern. 47 (12) (2017) 4014–4024.
[13] C. Lu, J. Jia, C.-K. Tang, Range-sample depth feature for action recognition, in: CVPR, 2014, pp. 772–779.
[14] J. Wang, Z. Liu, Y. Wu, J. Yuan, Mining actionlet ensemble for action recognition with depth cameras, in: CVPR, 2012, pp. 1290–1297.
[15] Y. Kong, Y. Fu, Bilinear heterogeneous information machine for RGB-D action recognition, IJCV 123 (2015) 1–22.
[16] R. Vemulapalli, F. Arrate, R. Chellappa, Human action recognition by representing 3d skeletons as points in a lie group, in: CVPR, 2014, pp. 588–595.
[17] V. Veeriah, N. Zhuang, G.-J. Qi, Differential recurrent neural networks for action recognition, in: ICCV, 2015, pp. 4041–4049.
[18] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, X. Xie, et al., Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks, in: AAAI, 2016, pp. 1–8.
[19] L. Chen, J. Lu, Z. Song, J. Zhou, Learning principal orientations descriptor for action recognition, in: ACPR, 2017, pp. 1–6.
[20] Y. Yuan, X. Zheng, X. Lu, A discriminative representation for human action recognition, PR 59 (2016) 88–97.
[21] J. Cho, M. Lee, H.J. Chang, S. Oh, Robust action recognition using local motion and group sparsity, PR 47 (2014) 1813–1825.
[22] M. Bregonzio, T. Xiang, S. Gong, Fusing appearance and distribution information of interest points for action recognition, PR 45 (3) (2012) 1220–1234.
[23] M. Barnachon, S. Bouakaz, B. Boufama, E. Guillou, Ongoing human action recognition with motion capture, PR 47 (1) (2014) 238–247.
[24] H. Wang, C. Yuan, W. Hu, C. Sun, Supervised class-specific dictionary learning for sparse modeling in action recognition, PR 45 (11) (2012) 3902–3911.
[25] L. Liu, L. Shao, P. Rockett, Boosted key-frame selection and correlated pyramidal motion-feature representation for human action recognition, PR 46 (7) (2013) 1810–1818.
[26] E.P. Ijjina, K.M. Chalavadi, Human action recognition using genetic algorithms and convolutional neural networks, PR 59 (2016) 199–212.
[27] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, L. Van Gool, Temporal segment networks: towards good practices for deep action recognition, in: ECCV, 2016, pp. 20–36.
[28] D. Tran, L. Bourdev, R. Fergus, L. Torresani, M. Paluri, Learning spatiotemporal features with 3d convolutional networks, in: ICCV, 2015, pp. 4489–4497.
[29] J. Zhang, W. Li, P.O. Ogunbona, P. Wang, C. Tang, RGB-D-based action recognition datasets: a survey, PR 60 (2016) 86–105.
[30] Y.-P. Hsu, C. Liu, T.-Y. Chen, L.-C. Fu, Online view-invariant human action recognition using RGB-D spatio-temporal matrix, PR 60 (2016) 215–226.
[31] H. Rahmani, A. Mahmood, D. Huynh, A. Mian, Histogram of oriented principal components for cross-view action recognition, TPAMI 38 (12) (2016) 2430–2443.
[32] J.F. Hu, W.S. Zheng, J. Lai, J. Zhang, Jointly learning heterogeneous features for RGB-D activity recognition, in: CVPR, 2015, pp. 5344–5352.
[33] M. Liu, H. Liu, C. Chen, Enhanced skeleton visualization for view invariant human action recognition, PR 68 (2017) 346–362.
[34] J. Wang, Y. Wu, Learning maximum margin temporal warping for action recognition, in: ICCV, 2013, pp. 2688–2695.
[35] K. Soomro, A.R. Zamir, M. Shah, Ucf101: a dataset of 101 human actions classes from videos in the wild, arXiv:/1212.0402 (2012).
[36] O. Oreifej, Z. Liu, Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences, in: CVPR, 2013, pp. 716–723.
[37] E. Vig, M. Dorr, D. Cox, Space-variant descriptor sampling for action recognition based on saliency and eye movements, in: ECCV, 2012, pp. 84–97.
[38] Y.-G. Jiang, Q. Dai, X. Xue, W. Liu, C.-W. Ngo, Trajectory-based modeling of human actions with motion reference points, in: ECCV, 2012, pp. 425–438.
[39] S. Mathe, C. Sminchisescu, Dynamic eye movement datasets and learnt saliency models for visual action recognition, in: ECCV, 2012, pp. 842–856.
[40] M. Jain, H. Jegou, P. Bouthemy, Better exploiting motion for better action recognition, in: CVPR, 2013, pp. 2555–2562.
[41] X. Peng, L. Wang, X. Wang, Y. Qiao, Bag of visual words and fusion methods for action recognition: comprehensive study and good practice, CVIU 150 (2016) 109–125.
[42] N. Srivastava, E. Mansimov, R. Salakhudinov, Unsupervised learning of video representations using LSTMs, in: ICML, 2015, pp. 843–852.
[43] J. Carreira, A. Zisserman, Quo vadis, action recognition? A new model and the kinetics dataset, in: CVPR, 2017, pp. 4724–4733.
[44] Z. Cai, L. Wang, X. Peng, Y. Qiao, Multi-view super vector for action recognition, in: CVPR, 2014, pp. 596–603.
[45] A.-A. Liu, Y.-T. Su, W.-Z. Nie, M. Kankanhalli, Hierarchical clustering multi-task learning for joint human action grouping and recognition, TPAMI 39 (1) (2017) 102–114.
[46] L. Sun, K. Jia, D.-Y. Yeung, B.E. Shi, Human action recognition using factorized spatio-temporal convolutional networks, in: ICCV, 2015, pp. 4597–4605.
[47] X. Yang, P. Molchanov, J. Kautz, Multilayer and multimodal fusion of deep neural networks for video classification, in: ACM MM, 2016, pp. 978–987.
[48] A. Shahroudy, T.-T. Ng, Y. Gong, G. Wang, Deep multimodal feature analysis for action recognition in RGB+ D videos, TPAMI 1 (2017) 1–14.
[49] J. Wang, Z. Liu, Y. Wu, Learning Actionlet Ensemble for 3D Human Action Recognition, in: Human Action Recognition with Depth Cameras, 2014, pp. 11–40.
[50] A. Shahroudy, T.-T. Ng, Q. Yang, G. Wang, Multimodal multipart learning for action recognition in depth videos, TPAMI 38 (10) (2016) 2123–2129.

Lei Chen received the B.E. degree in Communication Engineering from Tianjin University, Tianjin, China, in 2013. He is currently a Ph.D. Candidate with the School of Electrical and Information Engineering, Tianjin University, China. His current research interests include deep learning, unsupervised learning, and action recognition.

Zhanjie Song received the B.S. degree in mathematics from Hebei University, Baoding, China, and the M.S. degree in mathematics from Hebei Normal University, Shijiazhuang, China, and the Ph.D. degree in probability theory and mathematical statistics, from the School of Mathematical Science, Nankai University, Tianjin, China, in 1988, 1999, and 2006, respectively. He spent 2002–2009 as the Director at Institute of Applied Mathematical, Department of Mathematics, Tianjin University, Tianjin, China (TUTC). He spent 2006–2008 as a Postdoctoral Fellow in signal and information processing, with School of Electrical and Information Engineering, TUTC, and as a Postdoctoral Fellow in ocean environment monitoring, with National Ocean Technology Center, Tianjin, China. He is currently a vice-director at Institute of TV and Image Information, and a Professor at School of Mathematics and School of Electrical and Information Engineering simultaneously, all in TUTC. His current research interests are in approximation of deterministic signals, reconstruction of random signals and statistical analysis of random processes.

Jiwen Lu received the B.Eng. degree in mechanical engineering and the M.Eng. degree in electrical engineering from the Xi'an University of Technology, Xi'an, China, and the Ph.D. degree in electrical engineering from the Nanyang Technological University, Singapore, in 2003, 2006, and 2012, respectively. He is currently an Associate Professor with the Department of Automation, Tsinghua University, Beijing,

China. His current research interests include computer vision, pattern recognition, and machine learning. He has authored or co-authored over 180 scientific papers in these areas, where more than 80 papers are published in the IEEE Transactions journals and top conferences such as CVPR, ICCV and NIPS. He serves as an Associate Editor of the IEEE Trans. on Circuits and Systems for Video Technology, Pattern Recognition, and the Journal of Visual Communication and Image Representation. He also served as an Associate Editor of the Pattern Recognition Letters, Neurocomputing and the IEEE Access. He is a member of the Multimedia Signal Processing Technical Committee and the Information Forensics and Security Technical Committee of the IEEE Signal Processing Society, and a member of the Multimedia Systems and Applications Technical Committee of the IEEE Circuits and Systems Society, respectively. He was a recipient of the National 1000 Young Talents Plan Program. He is a senior member of the IEEE.

Jie Zhou received the B.S. and M.S. degrees both from the Department of Mathematics, Nankai University, Tianjin, China, in 1990 and 1992, respectively, and the PhD degree from the Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology (HUST), Wuhan, China, in 1995. From 1995 to 1997, he served as a postdoctoral fellow in the Department of Automation, Tsinghua University, Beijing, China. Since 2003, he has been a full professor in the Department of Automation, Tsinghua University. His research interests include computer vision, pattern recognition, and image processing. In recent years, he has authored more than 200 papers in peer-reviewed journals and conferences. Among them, more than 50 papers have been published in top journals and conferences such as the IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Image Processing, and CVPR. He is an associate editor for the IEEE Trans. on Pattern Analysis and Machine Intelligence, the International Journal of Robotics and Automation and two other journals. He received the National Outstanding Youth Foundation of China Award. He is a senior member of the IEEE.