Computing and Informatics, Vol. 37, 2018, 1339–1362, doi: 10.4149/cai_2018_6_1339

LEARNED SPATIO-TEMPORAL TEXTURE DESCRIPTORS FOR RGB-D HUMAN ACTION RECOGNITION

Zhengyuan Zhai, Chunxiao Fan, Yue Ming

Beijing University of Posts and Telecommunications
Beijing Key Laboratory of Work Safety Intelligent Monitoring
Xitucheng Road 10
100 876 Beijing, China
e-mail: {zhaiyuan, cxfan@bupt.edu.cn}, myname35875235@126.com

Abstract. Due to the recent arrival of Kinect, action recognition with depth images has attracted researchers' wide attentions and various descriptors have been proposed, where Local Binary Patterns (LBP) texture descriptors possess the properties of appearance invariance. However, the LBP and its variants are most artificially-designed, demanding engineers' strong prior knowledge and not discriminative enough for recognition tasks. To this end, this paper develops compact spatio-temporal texture descriptors, i.e. 3D-compact LBP(3D-CLBP) and local depth patterns (3D-CLDP), for color and depth videos in the light of compact binary face descriptor learning in face recognition. Extensive experiments performed on three standard datasets, 3D Online Action, MSR Action Pairs and MSR Daily Activity 3D, demonstrate that our method is superior to most comparative methods in respects of performance and can capture spatial-temporal texture cues in videos.

Keywords: 3D pixel differences vectors, compact binary face descriptor, feature fusion, human action recognition, RGB-depth videos

Mathematics Subject Classification 2010: 68Txx

1 INTRODUCTION

As an important field of computer vision, human action recognition (HAR) has acquired many researchers' attention and been extensively used in our real life, such as human-computer interaction, smart video surveillance and assisted living. Researchers mainly focus on recognizing actions from common videos in the past, facing with the challenge of variable illuminations, cluttered background and partial occlusions. Lately, due to the prevalence of low-cost depth sensors like Kinect, color and depth data are more easily to access simultaneously. Complementary to color images, depth images are robust to the change in lighting conditions and background, also can provide 3D structure of the object. Color images can offer more color, texture and appearance information than depth images. Consequently, more researchers are paying attention to recognize actions with color and depth images, namely RGB-D action recognition [7, 12, 20, 27, 28].

It has been shown that texture features are always important descriptors for the task of recognition, like the Local Binary Pattern (LBP) [1]. However, there are two issues when employing such features to recognize action from depth images. For one thing, most texture feature descriptors [2, 3, 4, 5, 17] require engineers' strong prior knowledge to determine the threshold, which is a constant during the process of encoding texture information such that it does not work for different datasets. For another, we cannot obtain discriminative LBP feature from depth images due to the lack of texture information. To solve the mentioned problems, we respectively develop 3D-compact local binary patterns (3D-CLBP) and local depth patterns (3D-CLDP) descriptor for color and depth images in this paper built on the Compact Binary Face Descriptor (CBFD) learning [22], which allows us to learn local binary patterns from raw pixels automatically. Furthermore, two fusion features are presented for RGB-D action recognition with developed 3D-CLBP and 3D-CLDP descriptor, thus they can simultaneously possess the texture characteristic of color and depth data.

As depicted in Figure 1, both color and depth videos are divided into non-overlap space-time volumes. The pixel differences vectors (PDVs) of volumes with same spatial locations and through the overall time are first extracted, then used to learn a spatial projection. To obtain compact and robust binary coding, we project all PDVs into binary vectors by learned spatial projections, and aggregate those binary vectors to low-dimensional LBP features. We further improve the discriminability of those LBP features by jointly employing the sparse coding and spatial-temporal pyramid pooling. In the end, we adopt the feature-level and decision-level fusion to simultaneously capture texture cues from color and depth data.

The main contributions of our work can be summarized as below:

1. Considering that the difference between central pixel and neighboring pixels may change both spatially and temporally when motion occurs, we propose a method to extract PDVs from a spatio-temporal volume using the difference of the former and later frames' neighboring pixels (central pixel) to current frame's central pixel (neighboring pixels).
2. Our 3D-CLBP and 3D-CLDP descriptors are extensions of CBFD, which straightly learn spatial thresholds from raw spatio-temporal pixels, making them

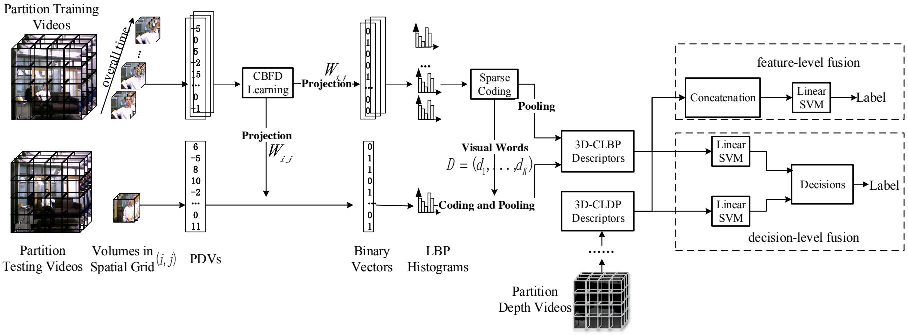


Figure 1. The pipeline of our developed method for action recognition with color and depth data. The pipeline presents the process of developing our 3D-CLBP descriptors for color videos. Likewise, we can develop our 3D-CLDP descriptors for videos in depth channel. In the end, two fusion techniques are used to achieve the task of action recognition.

more selective and suitable for different datasets than hand-crafted descriptors since we do not require engineers' strong prior knowledge.

3. We investigate different fusion methods to explore simultaneously preserving texture cues in color and depth data, further validate these methods in experiment to demonstrate the complementary nature of RGB and depth information in action recognition task.

The rest of our paper is organized as follows. The related work is given in Section 2. Section 3 describes detailed process of our compact binary codes learning in videos. The introduction of our classification and fusion methods are contained in Section 4. Section 5 reports the experiment parameters tuning and experiments' results on three datasets. Conclusions of this paper are given in Section 6.

2 RELATED WORK

This section first reviews some related researches on action recognition using LBP-like features from depth channel. We also briefly investigate some existing fusion works utilizing the data from color and depth channel.

Li et al. [6] first reported the work of action recognition from depth sequences. Afterwards, a variety of descriptors have been proposed, such as Spatio-Temporal features [8, 9, 11, 12, 20], Shape-motion features [13, 15, 16], and Texture descriptors [17, 18, 19]. Among those, texture descriptors are robust to subjects' clothing, appearance, and can capture substantial texture variations in the video. However, there is absence of abundant color and texture cues in depth images such that most extended LBP descriptors developed for color videos are not available in depth images sequence. To this end, the work [17] first projected an entire depth video into Depth Motion Maps (DMMs) from three projection views (front, side and top), then employed the LBP descriptor on these projected DMMs. Later, Bulbul [18] respectively calculated the LBP and Edge Oriented Histograms (EOHs) within overlapping and non-overlapping blocks on DMMs to extract local texture and dense shape information. The DMMs can provide adequate texture information for action recognition and well solve the variation of different videos' duration. Nevertheless, the work [19] indicated that the DMMs are presentation of entire depth video, cannot possess the motion and appearance information in temporal. Toward this problem, they extracted space-time auto-correlation of gradients as a complementary feature to conquer the loss of temporal information in the process of generating DMMs. Instead of computing LBP feature from DMMs, the work in [21] put forward the Gradient-LBP (G-LBP) descriptor to encode facial information from 2D depth images. The common fault of above works is that these existing LBP-like descriptors are all hand-crafted, which demands engineers' strong prior knowledge. To eliminate the defect of hand-crafted LBP descriptors, Lu [22] proposed a compact binary face descriptor (CBFD) learning for face recognition. With CBFD learning, obtained binary codes can evenly distribute at each bin and contain more discriminative information than hand-crafted descriptors. Enlightened by this, we attempt to extend the CBFD learning in 2D images to 3D videos in this paper.

In the light of complementary nature of color and depth information, some earlier works using both color and depth data in different recognition tasks can be found in those works. For instance, Ni et al. [7] derived Depth-layered multi-channel Spatio-Temporal Interest Points (STIPs) and 3D Motion History Images (3D-MHIs) from primitive STIPs and MHIs to fuse color and depth information for activity recognition. They also show a fusion framework to localize complex activity in videos by integrating information from grayscale and depth images in [23]. Zhu [24, 25] investigated some previous depth features developed for HAR, further combined these features and STIPs-based feature in color channel with various fusion schemes. Considering that features from RGB and depth channel share some similar structure, the works [26, 27, 28] explored the relationship between visual and depth features with different learning methods, which projected visual and depth features into a common subspace. The difference between them is that the work [26] projects Local Flux Feature (LFF) extracted from RGB and depth channel into a hamming subspace, while the works [27] and [28] learn the projection with label information. Similar to above learning methods, the works [29] and [30] respectively utilized graph-based genetic programming (RGGP) and regularized reconstruction independent component analysis deep network to build the relationship between RGB and depth modality for recognition tasks. Besides, Jia et al. [31] treated action data as fourth-order tensor, and discovered the correlation between RGB and depth modalities with cross-modality regularized transfer learning. Zhang and Parker [32] detected STIPs in saliency maps constructed by color and depth videos, then calculated 4-dimensional color-detph (CoDe4D) orientation histogram descriptor on each interest point. Results of above methods fully highlight that combining color and depth data can benefit to the task of action recognition, leading us to obtain selective representation for action recognition with our developed color and depth feature. Moreover, Kong and Fu [36] proposed the bilinear heterogeneous information machine (BHIM) to learn cross-modal features for RGB-D action recognition, which captures heterogeneous visual and depth information simultaneously. While Shahroudy et al. [41] utilized a deep autoencoder-based nonlinear common component analysis network to discover the shared and informative componets of RGB and depth data for an action. With the development of deep learning, many convolution neural networks (ConvNets) based methods were proposed for action recognition and obtained promising recogniton performance. Karpathy [37] and Tran [39] respectively utilize a deep 3-dimensional convolutional network (3D ConvNet) to recognize actions in video. Simonyan and Zisserman [38] proposed a two stream framework which employ two ConvNets to respectively extract features from appearance and motion streams then fuse the results for recognition. To deal with excessive computational cost when applied 3D ConvNet on long video sequences, the temporal segment network (TSN) [40] was proposed based on long-range temporal structure modeling. However, those ConvNets based methods need a great deal of training samples and have no evident advantage over traditional methods for our concerned small RGB-D datasets in this paper.

3 COMPACT BINARY CODES LEARNING

We first elaborate the motivation of our 3D pixel difference vectors (PDVs) extraction, then present how to compute discriminative 3D-PDVs from a spatio-temporal volume in this section. In the end, we extend the Compact Binary Face Descriptor (CBFD) learning in face images to videos.

3.1 Motivation and 3D PDVs

For a video, previous methods always calculate the differences between central pixel (or neighboring pixels) of current frame and neighboring pixels (or central pixel) of frames before (after) it, then compare the differences of both to obtain binary values. Such methods can miss the information of current frame since the pixel of current frame will be easily counteract when it is greater or less than corresponding pixels before and after the current frame. Moreover, we observe that the pixel at central location (x, y) of current time t may shift to neighboring location $(x + \Delta x, y + \Delta y)$ at time $t - \Delta t$ or $t + \Delta t$ when motions happen and vice versa. This stimulates us to find a better way to capture textures dynamic changes during motions occurrence.

Let $\{V_n^c, V_n^d\}$, $n = 1, 2, \ldots, N_0$ be N_0 color-depth video pairs in the dataset. For a video V_n^c (or V_n^d), it's often divided into some space-time volumes $\{V_{ij,k}^c, i = 1, \ldots, M; j = 1, \ldots, N; k = 1, \ldots, F\}$ of fixed size, like $20 \times 20 \times 5$ in our experiment, by $M \times N \times F$ grids. To obtain discriminative pixel differences vector $PDV_{ij,k}$ of a spatio-temporal volume $V_{ij,k}^c$, our method not only calculates the differences between current frames' central pixel and the former (latter) frames' neighboring pixels, also measures the differences between the former (latter) frames' central pixel

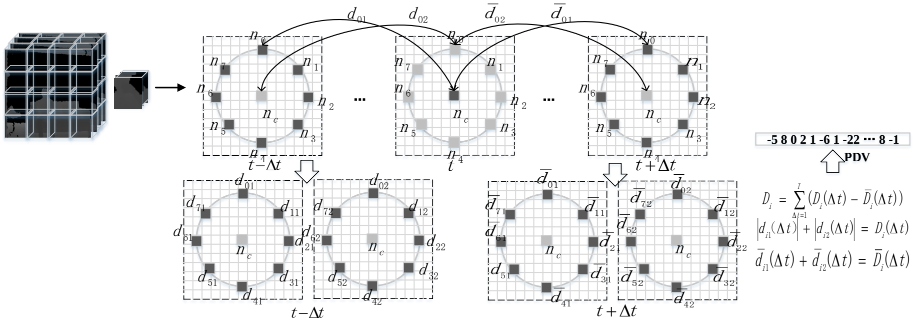

d_{01}
d_{02}
$\overline{d}_{02}$
$\overline{d}_{01}$
n_0
n_1
n_2
n_3
n_4
n_5
n_6
n_7
n_c
$t-\Delta t$
t
$t+\Delta t$
-5 8 0 2 1 -6 1 -22 ⋯ 8 -1
PDV
$D_i = \sum_{\Delta t=1}^{T}(D_i(\Delta t) - \overline{D}_i(\Delta t))$
$|d_{i1}(\Delta t)| + |d_{i2}(\Delta t)| = D_i(\Delta t)$
$\overline{d}_{i1}(\Delta t) + \overline{d}_{i2}(\Delta t) = \overline{D}_i(\Delta t)$
d_{71} d_{11} d_{61} d_{21} d_{51} d_{31} d_{41}
d_{72} d_{12} d_{62} d_{22} d_{52} d_{32} d_{42}
$\overline{d}_{71}$ $\overline{d}_{11}$ $\overline{d}_{61}$ $\overline{d}_{21}$ $\overline{d}_{51}$ $\overline{d}_{31}$ $\overline{d}_{41}$
$\overline{d}_{72}$ $\overline{d}_{12}$ $\overline{d}_{62}$ $\overline{d}_{22}$ $\overline{d}_{52}$ $\overline{d}_{32}$ $\overline{d}_{42}$

Figure 2. Illustration of our method to extract PDVs from a volume both spatially and temporally. Here we just show the process of calculating the frames at time $t - \Delta t$ and $t + \Delta t$ to current frame at time t. The same process can be used for depth data after computing the gradient of each frame.

and current frames' neighboring pixels, so that it can better capture the dynamic change of texture both spatially and temporally.

As depicted in Figure 2, given a $20 \times 20 \times 5$ volume V, central point n_c and its neighbors n_i, $i = 0, \ldots, 7$ at time t, $t - \Delta t$ and $t + \Delta t$, $\Delta t = 1, 2$. We first compute the pixel differences between n_c at time t and n_i at time $t - \Delta t$, $t + \Delta t$, respectively written as

$$d_{i1}(\Delta t) = I_{n_c}(t) - I_{n_i}(t - \Delta t), \quad \overline{d_{i1}}(\Delta t) = I_{n_c}(t) - I_{n_i}(t + \Delta t), \quad \Delta t = 1, 2 \tag{1}$$

where $I_{n_c}(t)$ denotes as the pixel value of n_c at time t. Likewise, we measure the pixel differences between n_c at time $t - \Delta t, t + \Delta t$ and n_i at time t, which can be represented as

$$d_{i2}(\Delta t) = I_{n_i}(t) - I_{n_c}(t - \Delta t), \quad \overline{d_{i2}}(\Delta t) = I_{n_i}(t) - I_{n_c}(t + \Delta t), \quad \Delta t = 1, 2. \tag{2}$$

Upon obtaining $d_{i1}(\Delta t)$, $d_{i2}(\Delta t)$, $\overline{d_{i1}}(\Delta t)$, $\overline{d_{i2}}(\Delta t)$, $i = 0, \ldots, 7$ at time $t - \Delta t$, $t + \Delta t, \Delta t = 1, 2$, the distance of current frame to frames before and after it, respectively denoted as D_i and $\overline{D_i}$, can be defined following

$$D_i = \sum_{\Delta t=1}^{2} (|d_{i1}(\Delta t)| + |d_{i2}(\Delta t)|), \quad \overline{D_i} = \sum_{\Delta t=1}^{2} (|\overline{d_{i1}}(\Delta t)| + |\overline{d_{i2}}(\Delta t)|). \tag{3}$$

Then, the difference between D_i and $\overline{D_i}$ of each pixel in volume V forms the final PDV. In respect of depth videos, we first compute the gradient information of each frame, then extract PDVs from depth videos with Equations (1), (2) and (3).

3.2 CBFD Learning

The CBFD has been proven to be more effective than conventional LBP descriptors in face recognition due to learning binary codes automatically instead of manually designing an encoding method. Herein, we extend CBFD learning in 2D images to 3D videos. Given all PDVs $\{PDV_{ij,k}^{c_m}, m = 1, \ldots, N_1\}$ from N_1 training color videos, we aim to train $M \times N$ projections ω_{ij}, $i = 1, \ldots, M$, $j = 1, \ldots, N$. For different color videos c_m, the number of PDVs k with a spatial grid (i, j) may be different and large. To improve computation efficiency, we randomly sample some $PDV_{ij,k}^{c_m}$ to train a projection $\omega_{ij} \in \mathbb{R}^{8\times 8}$. For simplicity, training PDVs set for ω_{ij} denotes as $\{pdv_{ij,k}, k = 1, \ldots, N'\}$, $pdv_{ij,k} \in \mathbb{R}^{8\times d}$, it can be projected and quantized to binary codes $b_{ij,k} = \left[b_{ij,k}^{(1)}, \ldots, b_{ij,k}^{(8)}\right]$, with $b_{ij,k}^{(8)} \in \{0, 1\}^{d\times 1}$ as below:

$$b_{ij,k} = 0.5 \times (\text{sgn}(pdv_{ij,k}^{\mathrm{T}} \omega_{ij}) + 1)$$

where sgn(x) is written as the sign function, equaling to 1 if $0 \leq x$ and -1 otherwise.

To get more discriminative and compact binary codes $b_{ij,k}$ for a volume, the CBFD learning imposes three important criterions (i.e. evenly distributed binary codes, less redundancy and less missed information in the learned binary codes) on objective function. The optimization objective function can be formulated as follows:

$$\begin{aligned}\min_{\omega_{ij}} J_1(\omega_{ij}) + \lambda_1 * J_2(\omega_{ij}) + \lambda_2 * J_3(\omega_{ij}) = & -\sum_{k=1}^{N'} \|b_{ij,k} - \mu_k\|^2 \\ & + \lambda_1 \sum_{k=1}^{N'} \left\|(b_{ij,k} - 0.5) - pdv_{ij,k}^{\mathrm{T}} \omega_{ij}\right\|^2 \\ & + \lambda_2 \left\|\sum_{k=1}^{N'} (b_{ij,k} - 0.5)\right\|^2 \qquad (4)\end{aligned}$$

In the above formula, N' indicates the number of PDVs extracted from training videos in spatial grid (i, j), μ_k serves as the mean of all training PDV's k^{th} binary code, updating in each iteration, λ_1 and λ_2 are two parameters to balance the effect of different terms in the objective function. The physical meaning of different terms and solution can be referred to [22].

4 CODING AND CLASSFICATION

In this section, we describe the idea of sparse coding and show how to aggregate learned LBP features into our final 3D-CLBP and 3D-CLDP descriptors based on the spatial-temporal pooling. We also introduce our classification and fusion techniques in this section.

4.1 Sparse Coding and Spatial-Temporal Pyramid Pooling

Instead of assigning each feature vector to the nearest visual word learned by k-means clustering in Bag of Words (BOW), the sparse coding enables a linear and sparse combination of all learned atoms in dictionary. This approach reduces the quantization error in the process of approximating a crude feature vector. To well keep more information of low-level features, the work [13] employs the coefficient-weighted differences between each visual word and a primitive feature vector. Consider a set of extracted features $P = (p_1, \dots, p_M)^{\mathrm{T}} \in \mathbb{R}^{M \times N}$ and coefficients of features $U = (u_1, \dots, u_M)^{\mathrm{T}} \in \mathbb{R}^{M \times K}$. The process of sparse coding can be represented by:

$$\min_{D,U} \sum_{m=1}^{M} \left(\left\| p_m - D^{\mathrm{T}} u_m \right\|^2 + \lambda \left\| u_m \right\|_1 \right), \quad \text{subject to } \left\| d_k \right\|^2 \leq 1, \quad \forall k = 1, \dots, K$$

where $D = (d_1, \dots, d_K)^{\mathrm{T}} \in \mathbb{R}^{K \times N}$is the learned dictionary with K visual words, and λ is the induced parameter of sparsity regularization. A detailed process of solving above optimization problem can be referred to [13]. Once learned the sparse coefficient u_{ik} of feature p_i to the k^{th} atom in dictionary, we can employ the coefficient-weighted difference $u_{ik}(p_i - d_k)$ as the coding.

In Section 3, we divide a video into some 3D volumes of fixed size, leading to different number of volumes with different videos. Hence, the final discriminative feature vector of different videos has different length, which cannot serve as the input of our classifier. The common solution is to perform spatial average pooling and temporal max (or sum) pooling for each video, which partition a video into various space-time grids. Let a space-time denoted by ST_t, which may be the entire video or a partitioned subsequence. vol_i is a small $20 \times 20 \times 5$ volume involved in ST_t and p_i is the feature of vol_i. $|ST_t|$ indicates the number of volumes in ST_t. Then the spatial average pooling and temporal max pooling with all partition are formulized as below:

$$v_k(t) = \frac{1}{|ST_t|} \sum_{vol_i \subset ST_t} u_{ik}(p_i - d_k), \quad v_k = \left(\max_t \{v_{k1}(t)\}, \dots, \max_t \{v_{kN}(t)\} \right).$$

We concatenate all pooled vectors v_k from K visual words to form the distinctive vector $V = (v_1, \dots, v_K)$ of KN dimensions as 3D-CLBP and 3D-CLDP descriptors.

4.2 Classification and Fusion

We employ widely used Support Vector Machine (SVM) as our action classifying framework. To deal with our sparse data, we use SVM with linear kernel as our classifier. The solver of linear SVM can be given in LIBLINEAR [33]. For data fusion, there are two ways: early fusion (sensor and feature level) and late fusion (rank, score, and decision level). We apply feature-level fusion and decision-level fusion to combine our 3D-CLBP and 3D-CLDP descriptors.

1. Feature-level fusion. We simply stack the 3D-CLBP and 3D-CLDP descriptor into a composite vector for classification. To reduce the complexity, we perform PCA with all 3D-CLBP and 3D-CLDP descriptors before concatenation.

2. Decision-level fusion. Different from the straightforward fusion, the decision-level fusion considers each of our feature as the input to a SVM classifier, then merges the results using the confidence scores generated by two individual SVM classifiers. Denoting ${f_q(x)}_k$ as the q^{th} classifiers' confidence scores predicting x to the k^{th} label, the posterior probability associated with q^{th} classifier $p_q(y_k|x)$ can be written as below:

$$p_q(y_k|x) = \frac{1}{1 + \exp(-{f_q(x)}_k)}.$$

Then we employ different decision rules to combine the two classifiers' results, such as Sum rule, Maximum rule shown in [25]. For Sum rule, Product rule, Maximum rule and Minimum rule, we respectively assign the final label y^* to the k^{th} label as Equations (5):

$$\begin{aligned}
P(y_k|x) &= \sum_{q=1}^{2} \alpha_q p_q(y_k|x), \\
P(y_k|x) &= \prod_{q=1}^{2} p_q(y_k|x)^{\alpha_q}, \\
P(y_k|x) &= \max_{q=1,2} p_q(y_k|x), \qquad (5) \\
P(y_k|x) &= \min_{q=1,2} p_q(y_k|x), \\
y^* &= \operatorname*{argmax}_{k=1,2,\ldots,C} P(y_k|x).
\end{aligned}$$

5 EXPERIMENTAL

In this section, we conduct experiments on three common RGB-D action datasets, with our developed 3D-CLBP and 3D-CLDP feature, and two fusion features. We first introduce the datasets and their settings, then illustrate experiments setup including parameter setting and tuning. Finally, we compare our proposed methods with other state-of-the-art methods.

5.1 Evaluation Datasets

We use three RGB-D action databases and their settings following previous work to evaluate the performance of our developed features, i.e. 3D Online Action, MSR Action Pairs, and MSR Daily Activity. Some sampled video frames of these datasets are illustrated in Figure 3.

3D Online Action (3Donline) dataset is an RGB-D action dataset including 7 human-object interactions in the living room: drinking, eating, using laptop, reading cellphone, making phone call, reading book and using remote, where each action is performed by 16 subjects twice. The dataset has three parts: same-environment, cross-environment and multiple unsegmented actions. We evaluate our method with same-environment actions and follow the experiment setting employed in [14], which adopts the first 8 subjects as training and the last 8 subject as testing.

The MSR Pairs Action dataset (MSRpairs) contains six pairs of actions: lift a box/place a box, pick up a box/put down a box, push a chair/pull a chair, put on a backpack/take off a backpack, stick a poster/remove a poster, and wear a hat/take off a hat. These paired-activities are performed by 10 subjects and each subject performs each activity 3 times. Thus, there are totally 720 color and depth sequences with the resolution of 480×640 and 240×320, respectively. The dataset is challenging since paired-activities are very similar but the motion happens in different temporal order. We employ the first five actors for training and the rest for testing as described in [15].

The MSR Daily Activity dataset (MSRdaily) totally has 720 sequences including color and depth (each 360) with 16 daily activities: drink, eat, read book, call cellphone, write on paper, use laptop, use vacuum cleaner, cheer up, sit still, toss paper, play game, lie down on sofa, walk, play guitar, stand up and sit down. In the dataset, RGB videos are offered with a resolution of 480×640, while each depth frame has a resolution of 240×320. Each daily activity is carried out by 10 subjects twice in the posture of standing or sitting, leading to large spatial and scaling changes. We follow the experiment setup in [16], taking subjects 1, 3, 5, 7, 9 as training and subjects 2, 4, 6, 8, 10 as testing.

5.2 Parameter Settings and Tuning

As aforementioned in Section 3, all videos (color and depth channel) are divided into multiple volumes, so we first adopt different volume sizes from $\{20\times20\times3, 20\times20\times5, 20\times20\times7, 40\times40\times3, 40\times40\times5, 40\times40\times7\}$ to test recognition performance over all experiment datasets, then employ the best volume size to tune other parameters. For the computation of neighboring pixels' differences we always choose 8 points with best neighborhood radius size R from $\{1, 2, 3\}$. Actually, we set $R = 1$ to get a fair result comparison with crude LBPs. λ_1 and λ_2 respectively denotes the weight of regularization term in Equation (5), we set λ_1 and λ_2 as the same value here and choose the best one from $\{0.0001, 0.001, 0.01, 0.1\}$ following [22]. To explore the impact of visual words size k in sparse coding, we tune k from $\{100, 200, 300, 400\}$

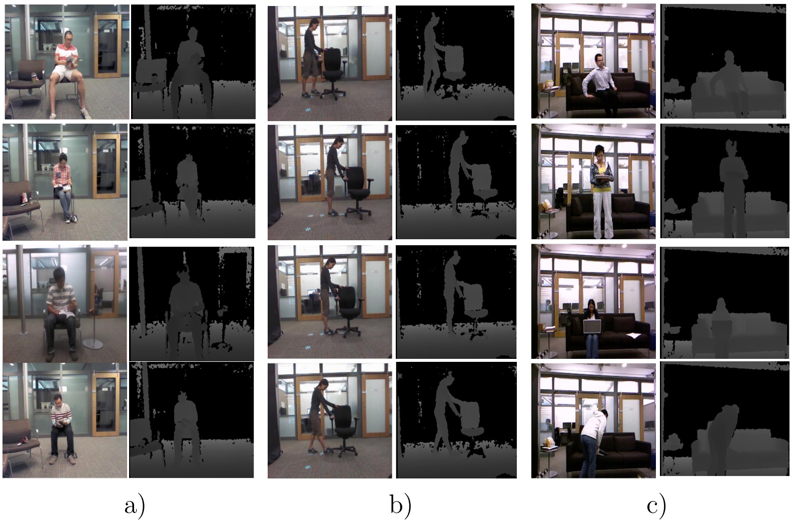
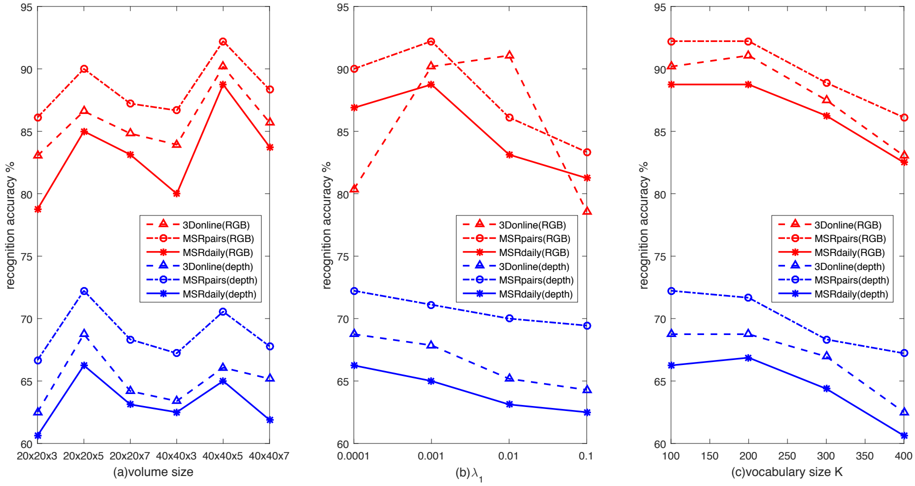


Figure 3. Some sampled color images and corresponding depth images from three experimental datasets: a) sampled frames of "read book" with different performers in 3D Online Action dataset, b) sampled frames of the "push a chair/pull a chair" pair from MSR Action Pairs dataset, c) sampled frames of activities with different performers' postures in MSR Daily Activity 3D dataset

with 10 fold-cross validations. The value of regularization parameter λ in sparse coding is set to 0.15 as [13]. Besides, we employ a space-time pyramid of $4 \times 3 \times 7$ grids to pool our features.

Figure 4 a) shows recognition performance with different volume size over all datasets (RGB and depth channel). It can be observed that our approach obtains the best recognition performance with volume size $40 \times 40 \times 5$ and $20 \times 20 \times 5$ respectively over all RGB and depth datasets since color and depth images respectively have resolutions of 480×640 and 240×320. Moreover, we find that the performance is more sensitive to the choice of volume size in temporal scale than spatial scale, and a moderate temporal size can lead to good recognition accuracy. The explanation here is that the small time interval cannot provide sufficient information for a motion, when the time interval is too long some local variations are missing. The recognition performance over experiment datasets using different λ_1 values is depicted in Figure 4 b). From this figure, we observe that the choice of best λ_1 value for RGB channel generally depends on the characteristic of dataset. For example, the 3Donline dataset with significant intra-class variations, like subjects' clothing and motion style, achieves the best recognition performance when using $\lambda_1 = 0.01$; the MSRpairs and MSRdaily datasets have little intra-class variations obtaining best performance with $\lambda_1 = 0.001$. Another interesting observation is that our approach over all depth datasets is not very sensitive to the value of λ_1,



a) b) c)

Figure 4. Performance comparison with different parameter values over all experiment datasets (RGB and depth channel) with our developed descriptors. a) performance comparison with different volume sizes, with $\lambda_1 = 0.001$ (RGB channel), 0.0001 (depth channel), $K = 100$; b) performance comparison with different λ_1 values, with volume size $= 40 \times 40 \times 5$ (RGB channel), $20 \times 20 \times 5$ (depth channel), $K = 100$; c) performance comparison with different K values, with volume size $= 40 \times 40 \times 5$, $\lambda_1 = 0.001$ (RGB channel), volume size $= 20 \times 20 \times 5$, $\lambda_1 = 0.0001$ (depth channel).

which intuitively illustrates the absence of texture information in depth images. We compare the recognition performance over all datasets with different number of visual words K in Figure 4 c). As presented in the figure, the performance over all datasets generally reaches best under a moderate vocabulary size, i.e. $K = 200$. Because small vocabulary size often leads to incorrectly assigning different features to a same visual word, and similar features are assigned to different visual words when vocabulary size is too large. However, we adopt $K = 100$ in our experiment since the performance is nearly the same when $K = 100$ and $K = 200$, but the dimension of feature will enlarge $59 \times 100 \times 84$, further resulting in higher computation complexity.

5.3 Experiment Results and Comparison

To evaluate our proposed method, we choose some previous representative RGB and depth features as our comparison, like Improved Dense Trajectories (IDTs)-based, DMMs-based features and STIPs-based fusion technique. Specifically, we implement our experiment over the ORGBD and MSRpair dataset using released source codes with default parameter settings for IDTs, DMMs and STIPs.

IDTs-based features [34] first sample feature points in each frame and track them with dense optical flow tracking technique. Then, some low-level feature (MBH, HOG or HOF) is encoded to IDTs-based feature with spatio-temporal pyramids, i.e. IDTs-MBH, IDTs-HOG, IDT-HOF.

DMMs-based features [17] including DMMs-LBP, DMMs-HOG and DMMs-EOH are built on depth motion maps generated by accumulating motion energy of projected depth maps from front, side and top view. Different from 3D features, DMMs-based features encode the motion characteristics of an action from 2D images.

STIPs-based features employ Harris 3D detector, cuboid detector or Hessian detector to detect interest points from videos or depth maps, and extract local feature descriptors from each detected interest point location. The common STIPs-based features are STIPs-HOG, STIPs-HOG3D and STIPs-HOF, etc.

In addition to the above features, we also investigate and report other methods using these three datasets. Detailed comparison results are presented in Table 1.

Channel + Methods + Classifier	Accuracy
RGB + IDTs-HOG/HOF + SVM [34]	77.68 %
RGB + STIPs-HOG + SVM [35]	79.46 %
Depth + depthHarris3D-DCSF + SVM [11]	61.70 %
Depth + Ordelet + AdaBoosting [14]	71.40 %
Depth + DMMs-LBP + KELM [17]	69.64 %
Both + STIPs-HOG3D + SVM [25]	91.07 %
Both + DSSCA-SSLM [41]	94.6 %
RGB + 3D-LBP + linearSVM	62.50 %
RGB + 3D-CLBP + linearSVM	90.18 %
Depth + 3D-LDP + linearSVM	37.50 %
Depth + 3D-CLDP + linearSVM	68.75 %
Both + feature-level fusion + SVM	93.75 %
Both + SVM + decesion-level(Sum) fusion	88.39 %
Both + SVM + decesion-level(Maximum) fusion	89.29 %
our 3D-CLBP + IDTs-FV + decesion-level fusion	**94.64** %

Table 1. Average recognition accuracy comparison of our method and previous approaches over the 3Donline dataset

5.3.1 Experiments Results and Comparison on 3Donline

The average recognition accuracy of feature-level fusion with our 3D-CLBP and 3D-CLDP descriptor achieves 93.75 % over the 3Donline dataset, as shown in Table 1. We compare our proposed method with some baseline methods, like IDTs-based, STIPs-based and DMMs-based methods in Table 1. The table presents that our 3D-CLBP descriptor with linearSVM in RGB channel obtains the average accuracy of 90.18 % significantly outperforming the IDTs-HOG/HOF and STIPs-HOG features. Although the average recognition accuracy of our 3D-CLDP descriptor

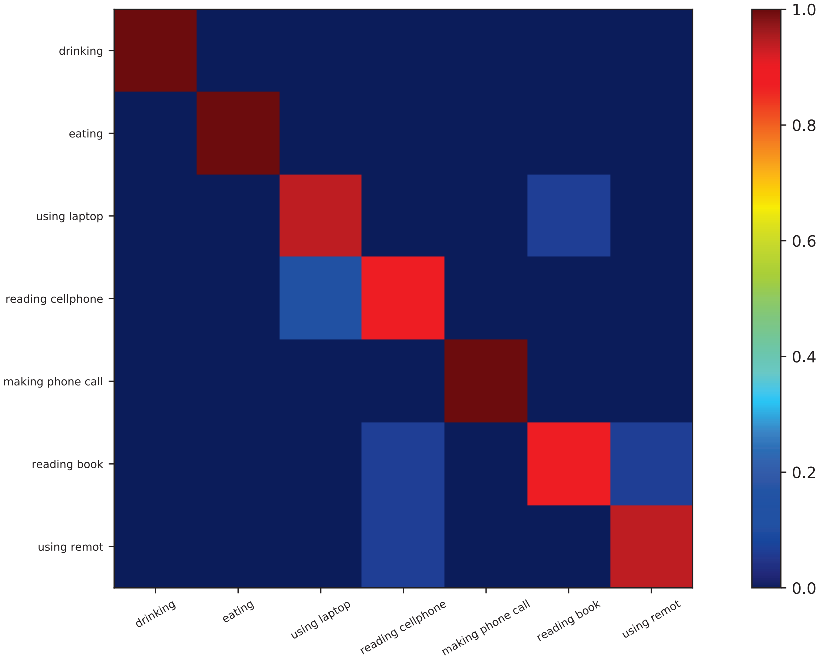


Figure 5. Fusion recognition results over the 3Donline dataset with our method

with linearSVM in depth channel is slightly below the Ordelet and DMMs-LBP method, it is higher than depth Harris3D-based method since Harris3D cannot detect sufficient spatio-temporal interest points in depth channel for partial occlusion in actions "using laptop", "reading cellphone" and "reading book". To demonstrate the superior performance of developed 3D-CLBP and 3D-CLDP, we also conduct experiments using primitive 3D-LBP and 3D-LDP with pre-set threshold 20 and 0.5, which obtains average recognition accuracy of 62.50 % and 37.50 %. Moreover, the feature-level, decision-Sum and decision-Maximum fusion with 3D-CLBP and 3D-CLDP are compared with STIPs-based fusion technique in [25]. And we obtain a promising result 93.75 % with feature fusion, around 2.7 % more than the STIP-HOG3D feature. For decision fusion, we obtain better recognition accuracy of 89.29 % with Maximum rule than weighted Sum rule in that our 3D-CLDP descriptor cannot work well on this dataset. To further improve the performance of our method, we combine the classify results of our 3D-CLBP descriptor and IDT-features with Fisher Vector (FV), and achieves the best performance of 94.64 %. This recognition result is also obtained by the deep shared-specific component analysis (DSSCA) network with structured sparsity learning machine (SSLM), which employed the deep convolutional network to extract modality-specific components of the modalities. The experiment results with our 3D-CLBP features and IDT descriptors are shown in Figure 5. It is observed that our method can recognize the action "drinking", "eating" and "making phone call", where interactive objects have distinctive texture characteristic. While the action "reading cellphone" and "reading book" not having distinctive interactive objects are recognized with some wrong actions. Above observation indicates that our 3D-CLBP descriptor is capable of encoding texture information.

Channel + Methods + Classifier	Accuracy
RGB + IDTs-HOG/HOF + SVM [34]	**100** %
RGB + STIPs-HOG + SVM [35]	81.67 %
Depth + Skeleton-LOP + SVM [10]	63.33 %
Depth + SNV + linearSVM [13]	98.89 %
Depth + HON4d + SVM [15]	93.33 %
Depth + DMMs-LBP + KELM [17]	78.89 %
Both + STIPs-HOG3D + SVM [25]	95.0 %
Both + DRRL + linearSVM [28]	99.44 %
Both + BHIM [36]	**100** %
Both + DSSCA-SSLM [41]	**100** %
RGB + 3D-LBP + linearSVM	67.78 %
RGB + 3D-CLBP + linearSVM	92.22 %
RGB + 3D-LDP + linearSVM	45.0 %
Depth + 3D-CLDP + linearSVM	72.23 %
Both + feature-level fusion + SVM	92.78 %
Both + decesion-level(Sum) fusion + SVM	93.89 %
Both + decesion-level(Maximum) fusion + SVM	91.67 %
Both + decesion-level(Minimum) fusion + SVM	86.11 %
our 3D-CLBP + IDTs-FV + decesion-level fusion	97.78 %

Table 2. Comparison of average recognition accuracy on the MSRpairs dataset with our method and some works

5.3.2 Experiments Results and Comparison on MSRpairs

Table 2 shows comparison results of our proposed approach with previous baseline methods over the MSRpairs dataset. From this table, it can be seen that the IDTs-based method in RGB channel obtains best recognition accuracy of 100 %, because actions in MSRpairs dataset have distinct motion direction. The same reason can also account for good performance of Depth super normal vector (SNV) in [13]. When compared to most mehods in RGB and Depth channel, we cannot obtain more satisfactory recognition performances with 3D-CLBP and 3D-CLDP descriptor due to encoding local texture changes but being incapable of capturing holistic changes in temporal-dependent sequences. Another observation is that the performance of 3D-CLDP descriptor reaches 72.23 % lower than DMMs-LBP's 78.89 %, which explicitly illustrates DMMs can better encode local texture than depth images. Nevertheless, the recognition performance of developed features are still far above

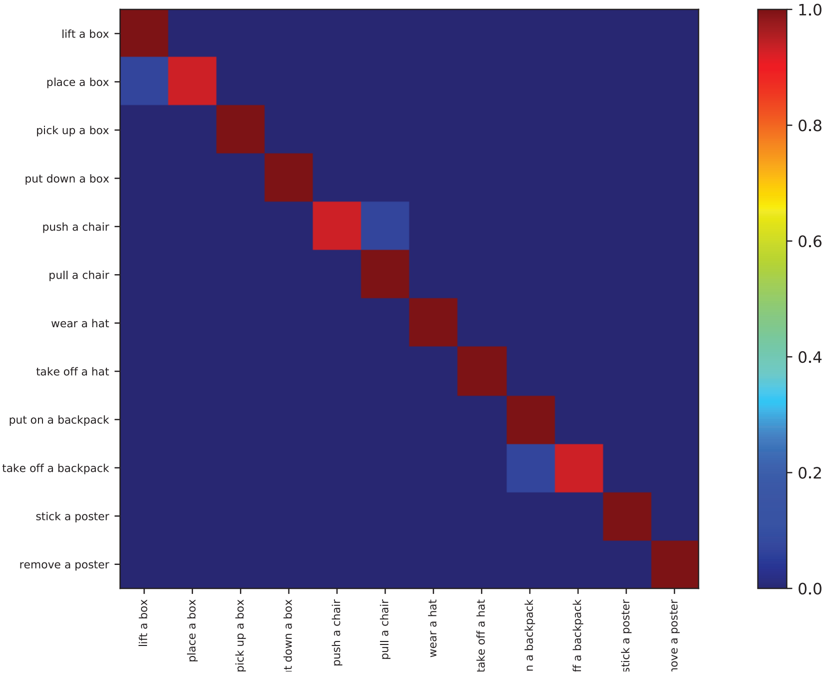


Figure 6. Obtained fusion recognition results over the MSRpairs dataset with our method

crude 3D-LBP and 3D-LDP features. Towards fusion methods using both RGB and depth data, the recogniton accuracy of proposed 3D-CLBP and 3D-CLDP with different fusion strategies reaches merely up to 93.89 %, which is lower than discriminative relational representation learning (DRRL), BHIM and lately DSSCA-SSLM. Even when fusing the results of our 3D-CLBP and effective IDTs-FV method, the fusion methods still cannot recognize all pairs actions which can be realized by just utilizing the IDTs method. It indicates that our proposed 3D-CLBP features with SVM mistake some actions with a larger probability. The reason is that some pairs actions in this dataset are extremely similar except the begining and end of a performed action, and they further generate same 3D PDVs with our proposed pixels difference computation method. This also accounts for the phenomenon that we do not obtain some encouraging experimental results using proposed method on this dataset comparing with the 3Donline dataset. In the end, we compare different decision fusion methods and gain better result with weighted Sum rule (balance parameter $\alpha = 0.8$ of two modality) than Maximum and Minimum rule. It well demonstrates that features in both RGB and Depth channel are important for the recognition task and have different impact. Figure 6 presents the experiment results by combining our 3D-CLDP with IDTs features and Fisher Vector. As shown in this figure, the fusion methods can recognize almost all pairs where temporal order and

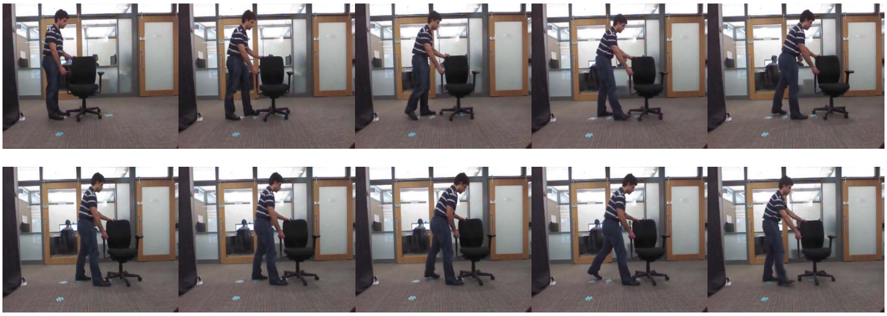


Figure 7. Sampled images except the begining and end of pair actions “push a chair” and “pull a chair” in MSRpairs dataset. Those images are very similar in terms of performed pose, leading to same PDVs and further incorrect identification

motion direction are significantly different, such as “pick up a box” and “put down a box” except some pair actions performed with some similar pose as illustrated in Figure 7.

5.3.3 Experiments Results and Comparison on MSRdaily

We compare the average recognition accuracy of our developed methods over the MSRdaily dataset with some baseline methods using color or depth images in Table 3. It can be observed that 3D-CLBP descriptor obtaining 88.75 % with RGB data is obviously superior to the performance of IDTs-based feature and an extended LTP, i.e. Center-Symmetric Motion LTP (CS-Mltp) feature [20], indicating our 3D-CLBP descriptor is more efficient than hand-crafted LTP. But the 3D-CLDP feature cannot get satisfactory result with depth data when compared with DMMs-LBP feature and some representative depth features, such as SNV, HON4d and Orderlet. To better evaluate the performance of our method over the MSRdaily dataset, we also compare our fusion features with some previous fusion methods, including STIPs-based fusion, LFF, DRRL, a deep learing model (RGGP) and BHIM methods. The results demonstrate that the feature-level and decesion-level fusion methods outperform those fusion approaches, increasing by 0.7 % to 6.9 %. However, the best performance of our fusion methods (namely feature-level fusion) only achieves 92.5 %, respectively 2.5 % and 5 % lower than the joint heterogeneous features learning (JOULE) model and lately DSSCA-SSLM method. This is because those two methods select discriminative features with learning models instead of merging features or recognition results directly in this paper. Another important observation in Table 3 is that our weighted decesion fusion features perform better than developed feature only using color or depth data, which highlights the importance of combing color and depth cues for recognition task. Besides, we analyze and compare fusion methods with different decision levels to explain the complementary nature of both modalities. The fused recognition results of this dataset with our 3D-CLBP and

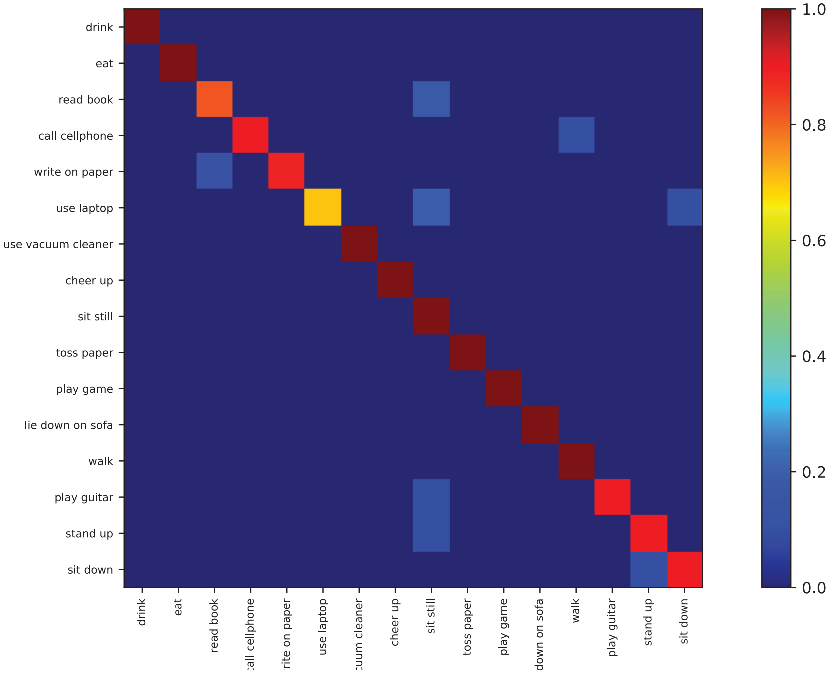


Figure 8. Fusion recognition results using proposed 3D-CLBP and IDTs-FV method over the MSRdaily dataset

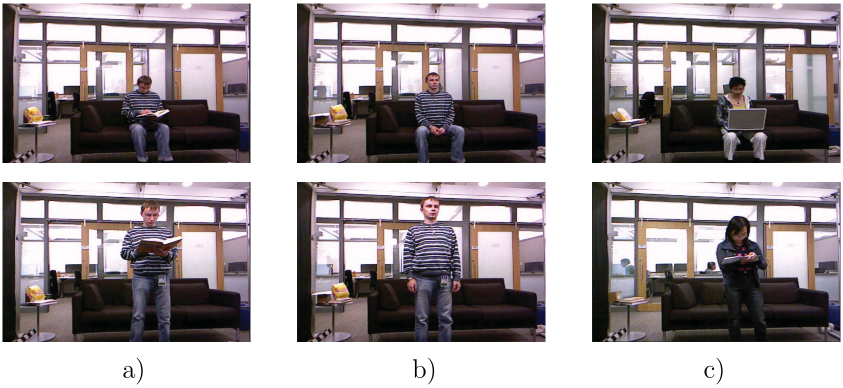


Figure 9. Incorrect identified actions from MSRdaily datasets. The action "read book" performed in sitting or standing way, i.e. column a), are identified as action "sit still" or "stand still", i.e. column b). Column c) enumerates two misidentified actions performed with seriously occlusion.

Channel + Methods + Classifier	Accuracy
RGB + CS-Mltp + SVM [20]	65.63 %
RGB + IDTs-HOG/HOF + SVM [34]	60.63 %
Depth + SNV + linearSVM [13]	86.25 %
Depth + HON4d + SVM [15]	80.00 %
Depth + Ordelet + SVM [16]	85.75 %
Depth + DMMs-LBP + KELM [17]	72.50 %
Both + STIPs-HOF/skeleton + 1NN [24]	89.29 %
Both + LFF-SPP + NN [26]	89.80 %
Both + DCP-DDP + JOULE-SVM [27]	95.0 %
Both + DRRL + linearSVM [28]	87.50 %
Both + RGGP(Deep model) [29]	85.60 %
Both + CoDe4D + SVM [32]	86.25 %
Both + BHIM [36]	86.88 %
Both + DSSCA-SSLM [41]	**97.5** %
RGB + 3D-CLBP + linearSVM	88.75 %
Depth + 3D-CLDP + linearSVM	66.25 %
Both + feature-level fusion + SVM	92.5 %
Both + decesion-level(Sum) fusion + SVM	90.63 %
Both + decesion-level(Product) fusion + SVM	91.87 %
our 3D-CLBP + IDTs-FV + decesion-level fusion	93.75 %

Table 3. Comparison of average recognition accuracy on the MSRdaily dataset using our method and other methods

IDTs-FV method is shown in Figure 8. From this figure, we can observe that our feature can correctly recognize all actions with acute motions, like "drink","eat", "cheer up","toss paper", "play game", "lie down on sofa","walk"and so on. But in terms of those actions with little motion such as "read book", "write on paper" and "play guitar", they may be wrongly identified as "sit still" in that those actions are performed in a sitting or standing way as shown in Figure 9 a). Thus, when those actions performed with no distinct pose, the calculated PDVs may be the same as "sit down still" or "stand up still" Figure 9 b). This demonstrates that our 3D-PDVs focus on the change of appearance rather than appearance information when motion happens. Besides, there are much wrong identification happen between "use laptop"and "sit still". The cause of this phenomenon can be interpreted as these actions are performed with seriously occlusion as demonstrated in Figure 9 c). We also notice that "sit still", "stand up"and "sit down" may be wrongly identified as other two actions since they share same action atomic.

6 CONCLUSIONS

This paper extends compact binary face descriptors learning in 2D images to 3D videos, which automatically learn discriminative binary representations for action recognition with color and depth videos. To this end, we develop a method to extract 3D pixels difference vectors (PDVs) from spatio-temporal volumes, then learn spatial projections with some PDVs extracted from the same spatial grid, and further project those PDVs into low-dimension binary codes. Moreover, we employ the sparse coding and spatial-temporal pooling to obtain discriminative representation of a video. In the end, we investigate different fusion methods to check the validity of combining color and depth data for action recognition. Extensive experiments performed on three standard benchmarks demonstrate that our method is superior to most methods being compared on 3D Online Action and MSR Daily Activity 3D datasets. However, we cannot obtain satisfying results on MSR Action Pairs dataset. Hence, combining the framework of deep learning or skeleton positions, to learn more discriminative descriptor for action recognition is our further research.

REFERENCES

[1] Ojala, T.—Pietikainen, M.—Maenpaa, T.: Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, 2002, No. 7, pp. 971–987, doi: 10.1109/tpami.2002.1017623.

[2] Kellokumpu, V.—Zhao, G.—Pietikäinen, M.: Human Activity Recognition Using a Dynamic Texture Based Method. In: Everingham, M., Needham, C. (Eds.): Proceedings of the British Machine Vision Conference (BMVC), Vol. 1, 2008, pp. 88.1–88.10, doi: 10.5244/c.22.88.

[3] Yeffet, L.—Wolf, L.: Local Trinary Patterns for Human Action Recognition. 2009 IEEE 12th International Conference on Computer Vision (ICCV '09), Kyoto, 2009, pp. 492–497, doi: 10.1109/iccv.2009.5459201.

[4] Mattivi, R.—Shao, L.: Human Action Recognition Using LBP-TOP as Sparse Spatio-Temporal Feature Descriptor. In: Jiang, X., Petkov, N. (Eds.): Computer Analysis of Images and Patterns (CAIP '09). Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 5702, 2009, pp. 740–747, doi: 10.1007/978-3-642-03767-2_90.

[5] Gupta, R.—Patil, H.—Mittal, A.: Robust Order-Based Methods for Feature Description. 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10), San Francisco, June 2010, pp. 334–341, doi: 10.1109/cvpr.2010.5540195.

[6] Li, W.—Zhang, Z.—Liu, Z.: Action Recognition Based on a Bag of 3D Points. 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition – Workshops (CVPRW '10), San Francisco, June 2010, pp. 9–14, doi: 10.1109/cvprw.2010.5543273.

[7] Ni, B.—Wang, G.—Moulin, P.: RGBD-HuDaAct: A Color-Depth Video Database for Human Daily Activity Recognition. 2011 IEEE International Conference on Computer Vision Workshops (ICCV 2011 Workshops), Barcelona, November 2011, pp. 1147–1153.

[8] Ming, Y.—Ruan, Q.—Hauptmann, A. G.: Activity Recognition from RGB-D Camera with 3D Local Spatio-Temporal Features. In: 2012 IEEE International Conference on Multimedia and Expo (ICME '12), Melbourne, July 2012, pp. 344–349, doi: 10.1109/icme.2012.8.

[9] Vieira, A. W.—Nascimento, E. R.—Oliveira, G. L.: STOP: Space-Time Occupancy Patterns for 3D Action Recognition from Depth Map Sequences. In: Alvarez, L., Mejail, M., Gomez, L., Jacobo, J. (Eds): Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications (CIARP 2012). Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 7441, 2012, pp. 252–259, doi: 10.1007/978-3-642-33275-3_31.

[10] Wang, J.—Liu, Z.—Wu, Y.—Yuan, J.: Mining Actionlet Ensemble for Action Recognition with Depth Cameras. Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12), Providence, June 2012, pp. 1290–1297, doi: 10.1109/cvpr.2012.6247813.

[11] Xia, L.—Aggarwal, J. K.: Spatio-Temporal Depth Cuboid Similarity Feature for Activity Recognition Using Depth Camera. Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '13), Portland, June 2013, pp. 2834–2841, doi: 10.1109/cvpr.2013.365.

[12] Koppula, H. S.—Saxena, A.: Learning Spatio-Temporal Structure from RGB-D Videos for Human Activity Detection and Anticipation. Proceedings of the 30th International Conference on Machine Learning – Volume 28 (ICML '13), Atlanta, June 2013, pp. 792–800, doi: 10.1177/0278364913478446.

[13] Yang, X.—Tian, Y. L.: Super Normal Vector for Activity Recognition Using Depth Sequences. Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14), Columbus, June 2014, pp. 804–811, doi: 10.1109/cvpr.2014.108.

[14] Yu, G.—Liu, Z.—Yuan, J.: Discriminative Orderlet Mining for Real-Time Recognition of Human-Object Interaction. In: Cremers, D., Reid, I., Saito, H., Yang, M. H. (Eds): Computer Vision – ACCV 2014. Springer, Cham, Lecture Notes in Computer Science, Vol. 9007, 2014, pp. 50–65, doi: 10.1007/978-3-319-16814-2_4.

[15] Oreifej, O.—Liu, Z.: HON4D: Histogram of Oriented 4D Normals for Activity Recognition from Depth Sequences. Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '13), Portland, June 2013, pp. 716–723, doi: 10.1109/cvpr.2013.98.

[16] Wang, J.—Liu, Z.—Wu, Y.—Yuan, J.: Learning Actionlet Ensemble for 3D Human Action Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 36, 2014, No. 5, pp. 914–927, doi: 10.1109/tpami.2013.198.

[17] Chen, C.—Jafari, R.—Kehtarnavaz, N.: Action Recognition from Depth Sequences Using Depth Motion Maps-Based Local Binary Patterns. 2015 IEEE Winter Conference Applications of Computer Vision (WACV '15), Waikoloa, January 2015, pp. 1092–1099, doi: 10.1109/wacv.2015.150.

[18] Bulbul, M. F.—Jiang, Y.—Ma, J.: DMMs-Based Multiple Features Fusion for Human Action Recognition. International Journal of Multimedia Data Engineering and Management (IJMDEM), Vol. 6, 2015, No.4, pp. 23–39, doi: 10.4018/ijmdem.2015100102.

[19] Chen, C.—Liu, M.—Zhang, B.—Han, J.—Jiang, J.—Liu, H.: 3D Action Recognition Using Multi-Temporal Depth Motion Maps and Fisher Vector. Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI '16), New York, July 2016, pp. 3331–3337.

[20] Luo, J.—Wang, W.—Qi, H.: Spatio-Temporal Feature Extraction and Representation for RGB-D Human Action Recognition. Pattern Recognition Letters, Vol. 50, 2014, No. 3, pp. 139–148, doi: 10.1016/j.patrec.2014.03.024.

[21] Huynh, T.—Min, R.—Dugelay, J. L.: An Efficient LBP-Based Descriptor for Facial Depth Images Applied to Gender Recognition Using RGB-D Face Data. In: Park, J. I., Kim, J. (Eds.): Computer Vision – ACCV 2012 Workshops. Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 7728, 2012, pp. 133–145, doi: 10.1007/978-3-642-37410-4_12.

[22] Lu, J.—Liong, V. E.—Zhou, X.—Zhou, J.: Learning Compact Binary Face Descriptor for Face Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 37, 2015, No. 10, pp. 2041–2056, doi: 10.1109/tpami.2015.2408359.

[23] Ni, B.—Pei, Y.—Moulin, P.—Yan, S.: Multilevel Depth and Image Fusion for Human Activity Detection. IEEE Transactions on Cybernetics, Vol. 43, 2013, No. 5, pp. 1383–1394, doi: 10.1109/tcyb.2013.2276433.

[24] Zhu, Y.—Chen, W.—Guo, G.: Evaluating Spatiotemporal Interest Point Features for Depth-Based Action Recognition. Image and Vision Computing, Vol. 32, 2014, No. 8, pp. 453–464, doi: 10.1016/j.imavis.2014.04.005.

[25] Zhu, Y.—Chen, W.—Guo, G.: Fusing Multiple Features for Depth-Based Action Recognition. ACM Transactions on Intelligent Systems and Technology (TIST), Vol. 6, 2015, No. 2, Art. No. 18, doi: 10.1145/2629483.

[26] Yu, M.—Liu, L.—Shao, L.: Structure-Preserving Binary Representations for RGB-D Action Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 38, 2016, No. 8, pp. 1651–1664, doi: 10.1109/tpami.2015.2491925.

[27] Hu, J.-F.—Zheng, W.-S.—Lai, J.—Zhang, J.: Jointly Learning Heterogeneous Features for RGB-D Activity Recognition. Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '15), Boston, June 2015, pp. 5344–5352.

[28] Kong, Y.—Fu, Y.: Discriminative Relational Representation Learning for RGB-D Action Recognition. IEEE Transactions on Image Processing, Vol. 25, 2016, No. 6, pp. 2856–2865, doi: 10.1109/tip.2016.2556940.

[29] Liu, L.—Shao L.: Learning Discriminative Representations from RGB-D Video Data. Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (AAAI '13), Bellevue, July 2013, pp. 1493–1500.

[30] Wang, A.—Lu, J.—Cai, J.—Cham, T.-J.—Wang, G.: Large-Margin Multi-Modal Deep Learning for RGB-D Object Recognition. IEEE Transactions on Multimedia, Vol. 17, 2015, No. 11, pp. 1887–1898, doi: 10.1109/tmm.2015.2476655.

[31] Jia, C.—Kong, Y.—Ding, Z.—Fu, Y. R.: Latent Tensor Transfer Learning for RGB-D Action Recognition. Proceedings of the 22nd ACM International Conference on Multimedia (MM '14), Orlando, November 2014, pp. 87–96, doi: 10.1145/2647868.2654928.

[32] Zhang, H.—Parker, L. E.: CoDe4D: Color-Depth Local Spatio-Temporal Features for Human Activity Recognition from RGB-D Videos. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 26, 2016, No. 3, pp. 541–555, doi: 10.1109/tcsvt.2014.2376139.

[33] Fan, R.-E.—Chang, K.-W—Hsieh, C.-J.—Wang, X.-R.—Lin, C.-J.: LIBLINEAR: A Library for Large Linear Classification. Journal of Machine Learning Research, Vol. 9, 2008, No. 9, pp. 1871–1874.

[34] Wang, H.—Kläser, A.—Schmid, C.—Liu, C.-L.: Dense Trajectories and Motion Boundary Descriptors for Action Recognition. International Journal of Computer Vision, Vol. 103, 2013, No. 1, pp. 60–79, doi: 10.1007/s11263-012-0594-8.

[35] Peng, X. J.—Wang, L. M.—Wang, X.—Qiao, Y.: Bag of Visual Words and Fusion Methods for Action Recognition: Comprehensive Study and Good Practice. Computer Vision and Image Understanding, Vol. 150, 2016, No. 3, pp. 109–125, doi: 10.1016/j.cviu.2016.03.013.

[36] Kong, Y.—Fu, Y.: Bilinear Heterogeneous Information Machine for RGB-D Action Recognition. Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '15), Boston, June 2015, pp. 1054–1062, doi: 10.1109/cvpr.2015.7298708.

[37] Karpathy, A.—Toderici, G.—Shetty, S.—Leung, T.—Sukthankar, R.—Li, F.-F.: Large-Scale Video Classification with Convolutional Neural Networks. Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14), Columbus, June 2014, pp. 1752–1732, doi: 10.1109/cvpr.2014.223.

[38] Simonyan, K.—Zisserman, A.: Two-Stream Convolutional Networks for Action Recognition in Videos. Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS '14), Montreal, December 2014, pp. 568–576.

[39] Tran, D.—Bourdev, L.—Fergus, R.—Torresani, L.—Paluri, M.: Learning Spatiotemporal Features with 3D Convolutional Networks. Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV '15), Santiago, December 2015, pp. 4489–4497, doi: 10.1109/iccv.2015.510.

[40] Wang, L. M.—Xiong, Y. J.—Wang, Z.—Qiao, Y.—Lin, D. H.—Tang, X. O.—Van Gool, L.: Temporal Segment Networks: Towards Good Practices for Deep Action Recognition. Proceedings of the 14th European Conference on Computer Vision (ECCV '16), Amsterdam, October 2016, pp. 20–36, doi: 10.1007/978-3-319-46484-8_2.

[41] Shahroudy, A.—Ng, T. T.—Gong, Y. H.—Wang, G.: Deep Multimodal Feature Analysis for Action Recognition in RGB+D Videos. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 40, 2018, No. 5, pp. 1045–1058, doi: 10.1109/tpami.2017.2691321.

Zhengyuan Zhai is working toward his Ph.D. degree in the Beijing Key Laboratory of Work Safety Intelligent Monitoring at the Department of Electronic Engineering, Beijing University of Posts and Telecommunications. His research interests are computer vision, human activity recognition, machine learning, deep learning and privacy-preserving. He has some papers in international conferences and journals about these areas.

Chunxiao Fan is currently Professor and the Director of Center for Information Electronic and Intelligence System. She served as a member of ISO/IEC JTC1/SC6 WG9, ASN.1 (since 2006) and Chinese Sensor network working group. She also was elevated to evaluation expert of Beijing Scientific and Technical Academy Awards. Her research interests include heterogeneous media data analysis, internet of things, data mining, communication software, and so on. In recent years, she has been Director of several Nation Science Foundation Projects. She published more than 30 papers in international journals and conferences, authored and edited three books and authorized several patents for inventions.

Yue Ming received her B.Sc. degree in communication engineering, her M.Sc. degree in human-computer interaction engineering, and the Ph.D. degree in signal and information processing from Beijing Jiaotong University, China, in 2006, 2008, and 2013, respectively. She worked as a visiting scholar in Carnegie Mellon University, U.S., between 2010 and 2011. Since 2013, she has been working as a faculty member at Beijing University of Posts and Telecommunications. Her research interests are in the areas of biometrics, computer vision, computer graphics, information retrieval, pattern recognition, etc. She has authored more than 40 scientific papers in these areas.