SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

May 2014, Vol. 57 052112:1–052112:13
doi: 10.1007/s11432-013-4938-y

Learning a discriminative mid-level feature for action recognition

LIU CuiWei, PEI MingTao*, WU XinXiao, KONG Yu & JIA YunDe

Beijing Laboratory of Intelligent Information Technology, School of Computer Science,
Beijing Institute of Technology, Beijing 100081*, China*

Received July 27, 2013; accepted September 21, 2013; published online October 9, 2013

Abstract In this paper, we address the problem of recognizing human actions from videos. Most of the existing approaches employ low-level features (e.g., local features and global features) to represent an action video. However, algorithms based on low-level features are not robust to complex environments such as cluttered background, camera movement and illumination change. Therefore, we propose a novel random forest learning framework to construct a discriminative and informative mid-level feature from low-level features of densely sampled 3D cuboids. Each cuboid is classified by the corresponding random forests with a novel fusion scheme, and the cuboid's posterior probabilities of all categories are normalized to generate a histogram. After that, we obtain our mid-level feature by concatenating histograms of all the cuboids. Since a single low-level feature is not enough to capture the variations of human actions, multiple complementary low-level features (i.e., optical flow and histogram of gradient 3D features) are employed to describe 3D cuboids. Moreover, temporal context between local cuboids is exploited as another type of low-level feature. The above three low-level features (i.e., optical flow, histogram of gradient 3D features and temporal context) are effectively fused in the proposed learning framework. Finally, the mid-level feature is employed by a random forest classifier for robust action recognition. Experiments on the Weizmann, UCF sports, Ballet, and multi-view IXMAS datasets demonstrate that out mid-level feature learned from multiple low-level features can achieve a superior performance over state-of-the-art methods.

Keywords action recognition, mid-level feature, feature fusion, temporal context

Citation Liu C W, Pei M T, Wu X X, et al. Learning a discriminative mid-level feature for action recognition. Sci China Inf Sci, 2014, 57: 052112(13), doi: 10.1007/s11432-013-4938-y

1 Introduction

Human action recognition has wide applications in many fields such as intelligent video surveillance, video retrieval, and human-computer interaction. However, it still remains a challenging problem due to many factors such as background, viewpoint, camera movement, occlusion, and variation in performance.

How to represent an action video with discriminative features is critical in human action recognition. An expressive representation can provide sufficient and discriminative information to the classifier, and significantly improves the recognition performance. Low-level visual representations of an action can be roughly divided into two categories: global representation and local representation. Global descriptors [1–3] extract visual information from videos to describe the entire human figure. However, global

*Corresponding author (email: peimt@bit.edu.cn)

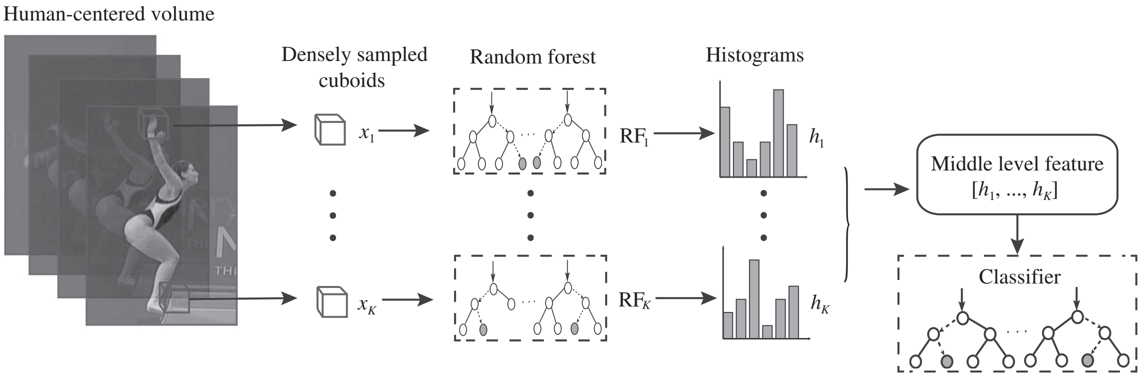


Figure 1 Overview of our method. Cuboids densely sampled from input volumes are represented by multiple low-level features (i.e., optical flow, HOG3D, and temporal context). A random forest is built for cuboids extracted at the same position of different human-centered volumes, and each cuboid obtains a histogram representing the posterior probabilities of all categories. Then the mid-level feature of a volume is obtained by concatenating the histograms of all the cuboids extracted in this volume. At last, another random forest is employed as the final classifier.

representation such as optical flow is sensitive to noise. Local representations [4–7] describe a video by extracting a set of primitive cuboids, each of which is calculated around a detected interest point in most cases. Local representation is more robust to noise, partial occlusion and global deformation than global representation, but it depends on the reliable detection of accurate and informative interest points. Since a single low-level descriptor is insufficient for the further recognition task, researchers try to create video representations with more discriminative power. One strategy is to merge multiple low-level descriptors into a unified framework. Recent studies [8,9] have shown that the fusion of multiple features (e.g., local feature and global feature) can benefit classification problems and achieve better recognition results, but they still rely on low-level features with limited robustness to action variations. Another group of methods [10–12] resort to transforming low-level descriptors into richer representations (called mid-level features) with more discriminative and descriptive ability. However, most of these approaches base their mid-level features on a single low-level feature to describe only one aspect of human actions.

In this paper, we propose a novel random forest learning framework to build a compact mid-level representation based on multiple low-level features for action recognition. Our method starts from densely sampled 3D cuboids of the input human-centered volumes, and characterizes each cuboid by both optical flow and histogram of gradient 3D (HOG3D) features. The optical flow feature captures the global motion information while the HOG3D descriptor describes the local appearance and motion. Then, each cuboid is classified by the corresponding random forest, and the posterior probabilities of all categories are normalized to a histogram. The histograms of all the cuboids extracted from a volume are concatenated to generate a compact mid-level representation of this volume. In this framework, the spatial context is utilized by building a separate random forest at each dense sampling spatial position. Moreover, we exploit the temporal context as another type of low-level feature for describing 3D cuboids, and then the optical flow, HOG3D and temporal features of cuboids are fused into a discriminative mid-level feature. Finally, another random forest is utilized to train our final classifier using the mid-level features. An overview of the proposed approach is illustrated in Figure 1.

The main contributions of this work include: (1) We propose a novel discriminative mid-level feature learned using random forest which achieves promising results on four public datasets. (2) Temporal context between 3D cuboids is exploited as a type of low-level feature for constructing the mid-level feature. (3) Multiple low-level features (i.e., optical flow, HOG3D, and temporal context) are effectively fused in the random forest learning framework to describe human actions from different perspectives. The proposed framework is general for merging any type of low-level features.

2 Related work

Recently, combination of multiple features has been investigated for human action recognition. Wang and Mori [9] integrated global motion descriptors and local part descriptors into a hidden conditional random field framework. Liu et al. [8] fused local spatio-temporal feature with spin-image feature for robust human action recognition.

The descriptive ability of low-level features is not enough, so some methods build mid-level representations derived from low-level features. Popular examples of mid-level representation of actions include bag-of-words [12–14], part-based model [10,15–17] and attributes [18]. Niebles et al. [12] quantized local spatio-temporal features to a code-book and utilized the bag-of-words model to create a sparse mid-level representation for action recognition. Then a lot of researches try to improve the bag-of-words model by building a discriminative code-book [13] or modeling the correlations among different visual words [14]. However, the above bag-of-words based approaches require reliable detection of interest points or densely sampling of the whole video, and do not consider the spatial and temporal context. Han et al. [10] employed a cascade CRF to recognize the motion patterns for both the entire body and each body part in a learned hierarchical manifold space. Wang et al. [15] presented a part based model to decompose an action into several parts to capture the local structure of the input data, and meanwhile they encoded pairwise relationships among different parts explicitly. Niebles et al. [16] extended the concept of part from a spatial region to a group of consecutive video frames to characterize the local temporal information of an action. Raptis et al. [17] developed a mid-level part model of individual spatio-temporal regions and their pairwise relations. For part based approaches, the accurate detection of body parts in various situations (such as occlusion) is still a crucial problem. In recent years, a semantic concept “attribute” is proposed to bridge the semantic gap between low-level features and high-level categories. Liu et al. [18] proposed a mid-level attribute-based representation which describes human actions as a set of attributes with semantic meaning, and this method needs to predefine the attributes and use additional training data to train the attribute classifier.

Our method is related to the work of Fathi and Mori [11]. They presented an algorithm to learn a mid-level motion feature from a signal low-level feature using AdaBoost, and their model is designed for two-class problem, so a multi-class classification problem has to be reduced into binary classification. Our mid-level feature built upon multiple low-level descriptors is natural for multi-class action recognition. Furthermore, spatial and temporal context is introduced into the random forest framework to learn our compact and semantic mid-level feature.

Our work is also relevant to random forest which is an ensemble of decision trees and classifies samples by majority voting over the outputs of all the trees. Random forest is built in a supervised manner, every non-leaf node splits into two children nodes depending on a certain property of training samples, while every leaf node is labeled as one class. Recently random forest has garnered growing interest in the computer vision community, and has been successfully applied to object recognition [19], action search [20], and human pose recognition [21]. Random forest has several appealing features. Firstly, random forest does not overfit as more trees are added, so it is good at processing our large high-dimensional data. Secondly, random forest is robust to noise since it treats each training datum equally by random selection; therefore our method can tolerate the jitter of input human figures. Thirdly, when building a tree, feature candidates are sampled randomly and independently, which provides a good manner to fuse multiple features. In addition, unlike binary classifiers, such as SVM and AdaBoost, random forest is capable of handling multi-class problems naturally, which makes it quite suitable for our method. Random forest is adopted twice in our method. Random forest is first used to learn a mid-level feature by fusing multiple low-level features, and then another random forest is utilized to train the final classifier.

3 Our method

We aim to learn a discriminative mid-level feature from multiple low-level features for human action recognition. In this work, a human body detector or tracker is required to extract the human-centered figures from a video. After stabilization, optical flow is calculated from the entire figure to characterize the global motion of the interested human. A video is segmented into a set of fixed-size sub-volumes for further processing. Then 3D cuboids are densely sampled from sub-volumes, each of which is a small spatio-temporal patch (e.g. $15 \times 15 \times 4$). A cuboid is represented by HOG3D feature computed at the local region and optical flow feature extracted from the global optical flow. Cuboids drawn at the same position of various sub-volumes are utilized to build a random forest, which generates a histogram of posterior probabilities for each cuboid. The concatenation of histograms of all cuboids extracted from a particular sub-volume is normalized to get the mid-level representation. Finally, we employ random forest for a second time to classify the input sub-volumes using the mid-level feature.

3.1 Low-level features

Two types of low-level features are merged to describe the motion and appearance of the interested human. The optical flow [1] feature is computed from the entire human figure to capture global motion information, and this motion descriptor shows favorable performance with noise, so it can tolerate the jitter of human figures caused by human detector or tracker, while the HOG3D [5] spatial-temporal descriptor is extracted from a single cuboid and characterizes the local motion and appearance information. The global motion feature is able to encode much information of action with powerful and abundant ability, whereas the local spatial-temporal feature can deal with noise and partial occlusion with good robustness. Thus the representation based on fusion of them is more robust to variations and more discriminative for classification.

Algorithm 1 Construction of mid-level feature.

```
Input:
    Training sub-volumes $\{v_s, s = 1 : S\}$;
Output:
    Mid-level features: $\{f(v_s), s = 1 : S\}$;
for s=1:S do
    Extract cuboids from sub-volume $v_s$ at K positions: $\{x_{s,k}, k = 1 : K\}$;
    Represent each cuboid with low-level features: $x_{s,k} = (x_{s,k}^{\mathrm{OF}}, x_{s,k}^{\mathrm{HOG3D}})$;
end for
for k=1:K do
    Build a random forest $\mathrm{RF}_k$, using cuboids $\{x_{s,k}, s = 1 : S\}$ sampled at position k;
end for
for s=1:S do
    for k=1:K do
        Classify cuboid $x_{s,k}$ with random forest $\mathrm{RF}_k$;
        The posterior probabilities of all classes are normalized to a histogram $\overline{h}(x_{s,k})$,
    end for
    Concatenate histograms of all the $K$ cuboids extracted from $v_s$ to obtain the
    mid-level feature: $f(v_s) = [\overline{h}(x_{s,1})^{\mathrm{T}}, \overline{h}(x_{s,2})^{\mathrm{T}}, \ldots, \overline{h}(x_{s,K})^{\mathrm{T}}]^{\mathrm{T}}$.
end for
return $\{f(v_s), s = 1 : S\}$;
```

3.2 Mid-level feature

The proposed random forest learning framework combines multiple low-level features to construct a mid-level representation with more discriminative and descriptive power. Suppose that each sub-volume generates a group of cuboids, denoted by $\{x_k = (x_k^{\mathrm{OF}}, x_k^{\mathrm{HOG3D}}), k = 1 : K\}$, where x_k^{OF} is the optical flow feature of the local cuboid x_k sampled from the global optical flow, and x_k^{HOG3D} represents the HOG3D feature calculated in the local cuboid x_k. At each sampling position, a collection of cuboids is extracted from all the training sub-volumes, and a random forest is built on these cuboids to learn the local mid-level representation which is a posterior probability histogram. Histograms of all the cuboids extracted from a sub-volume are concatenated to create our mid-level feature of this sub-volume. The construction process of our mid-level feature is summarized in Algorithm 1.

3.2.1 *Construction of trees*

Each tree in the random forest is independently grown from a bootstrap training set obtained by random sampling in the original training set, using CART methodology [22]. All the training samples are dropped down a tree from the root. In order to split a node and the training samples reaching the node, one type of low-level features is randomly selected, and then feature candidates are randomly sampled from this type of features. Especially, we utilize optical flow and HOG3D features. To this end, we use a predefined ratio γ regarded as the prior probability of optical flow and a random number $\delta \in [0,1]$ to co-determine which type of low-level features is selected. If δ is less than γ, then optical flow is used for node split; otherwise, HOG3D is selected. The node splits into two children nodes according to the feature candidate with the largest information gain, and each datum at this node is sent to one of the children nodes. The maximal depth of tree $dep_{\max}$ and the minimal number of samples for a parent node $par_{\min}$ are set to control the node split, so a node stops splitting in three states:

- All of the samples assigned to this node belong to the same class.
- The node has got to the limited depth of the tree.
- The number of samples assigned to this node is fewer than $par_{\min}$, which means the node does not have enough samples to split.

In the above three cases, a node is treated as a leaf. Instead of marking the leaf node by one of the class labels [22], we use a vector to store the distribution of class labels [23], $P = [p^1, p^2, ..., p^M]$, where M is the number of classes, and p^i is the posterior probability of data at the corresponding leaf node belonging to class i. The probability p_n^i of leaf node n is given by

$$p_n^i = p_n(y(x) = i) = \frac{N_i}{\sum_{j=1}^{M} N_j}, \tag{1}$$

where N_j represents the number of samples of class j arriving at node n. The posterior probability p_n^i is evaluated as the ratio of the number of samples of class i to the total number of samples reaching this node.

3.2.2 *Construction of mid-level features*

During testing, a sample x (i.e., a local cuboid) is passed down to each of the Tr trees. Suppose that the sample x reaches leaf node $l(tr,x)$ of tree tr , then x is classified to the class with the largest average of the posterior probabilities of all trees:

$$\widetilde{y}(x) = \arg\max_i P_i(x) = \arg\max_i \frac{1}{Tr} \sum_{tr=1}^{Tr} p_{l(tr,x)}^i, \tag{2}$$

where $P_i(x)$ is the average of the posterior probabilities of class i and constitutes a good local descriptor. For a testing sample x , the posterior probabilities of all classes $\{P_i(x)|i = 1 : M\}$ are normalized to a histogram $\overline{h}(x) = [\overline{P}_1(x), \overline{P}_2(x), ..., \overline{P}_M(x)]^{\mathrm{T}} \in \mathbb{R}^{M\times 1}$, which can be used as a mid-level representation of local cuboid x. However, there exists an over-fitting problem for training samples that posterior probability of the true class is very close to 1. Hence, mid-level representations of training samples created in this manner are quite different from those of testing samples, and a classifier built on these mid-level representations would not achieve good performance.

We introduce the out-of-bag estimate to construct the mid-level features for training samples. Out-of-bag estimate is a performance evaluation criterion characteristic of random forest. As described in Subsection 3.2.1, when building a tree, about 1/3 of the training samples are left out of the bootstrap training set. Accordingly, a training sample x is only trained on 2/3 trees in the forest, and the ensemble of trees not trained on x constitutes a new classifier, called the out-of-bag classifier. Then the out-of-bag estimate for the generalization error is the error rate of the out-of-bag classifier on the training set. According to Breiman [24], empirical evidence shows that the out-of-bag estimate is as accurate as that using a test set of the same size as the training set. Therefore, the out-of-bag estimate provides a good solution to the over-fitting problem of training samples. For a training sample x, the out-of-bag classifier substitutes for the whole forest to get a posterior probability histogram $\overline{h}(x)$.

The above process is done for each sampling position separately. Histograms of all the K cuboids extracted from an input sub-volume v are concatenated to the mid-level feature of v : $f(v) = [\overline{h}(x_1)^{\mathrm{T}}, \overline{h}(x_2)^{\mathrm{T}}, ..., \overline{h}(x_K)^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{(M\times K)\times 1}$.

Algorithm 2 Selection of a feature for node split.

```
Input:
    A set of training samples that reach the node: $\{x_k = (x_k^{\mathrm{OF}}, x_k^{\mathrm{HOG3D}}, x^{\mathrm{Temp}})\}$ ;
Output:
    One feature selected to split a node: $f_{\mathrm{sel}}$;
Initializing the feature candidate set FCS = $\emptyset$;
Generating a random number $\delta$ between 0 and 1;
if $\delta < \gamma$ then
    add $\mathrm{num_{opt}}$ random selected optical flow features to FCS;
else
    add $\mathrm{num_{hog3d}}$ random selected HOG3D features to FCS;
end if
Generating a random number $\theta$ between 0 and 1;
if $\theta < \tau$ then
    add two-dimensional temporal feature to FCS;
end if
maxgain = $-\infty$;
for each $fc \in$ FCS do
    computing information gain, gain($fc$);
    if gain($fc$) >maxgain then
        maxgain = gain($fc$);
        $f_{\mathrm{sel}} = fc$;
    end if
end for
return $f_{\mathrm{sel}}$;
```

3.3 Mid-level features with temporal context

An input sub-volume is comprised of several continuous human-centered figures of the original video. A group of cuboids are extracted from the sub-volume, and then each cuboid is processed by the corresponding random forest to generate a mid-level representation. At last, the concatenation of mid-level representations of all cuboids creates the mid-level feature of the input sub-volume. Since the cuboids are extracted at different spatial positions from a sub-volume, the mid-level feature described in Subsection 3.2 only considers the spatial information of cuboids. In this section, we introduce the temporal context between cuboids which is characterized by the relative temporal distance between pair-wise cuboids into the construction of mid-level features.

Suppose that an original video is a sequence of T frames, and a sub-volume is composed of L frames (frame t to frame $t + L - 1$) of this video, and the temporal context of this sub-volume is represented by a vector: temp_cxt= $[t/T, g(t/T)]^{\mathrm{T}}$. Here, t/T is the relative temporal position of the sub-volume in the whole video, and $g(t/T)$ denotes a set of functions of t/T. A variety of functions $g(t/T)$ can be exploited. Specifically, we use $g(t/T) = |t/T - 0.5|$ which describes the temporal distance between this sub-volume and the center of the video, and so we get a two-dimensional temporal context temp_cxt= $[t/T, |t/T - 0.5|]^{\mathrm{T}}$. For computational simplicity and efficiency, we set the temporal size of local cuboids the same as that of sub-volumes, so the temporal context of a cuboid is represented by that of the sub-volume from which it is extracted.

Our random forest based framework is general for fusing multiple low-level features, and we can treat the temporal context of local cuboids as a type of low-level feature. So a cuboid can be represented by optical flow, HOG3D, and temporal context, and denoted by $x = (x^{\mathrm{OF}}, x^{\mathrm{HOG3D}}, x^{\mathrm{Temp}})$. During the construction of a tree, a node splits into two children nodes according to one feature chosen from a set of randomly selected feature candidates. The procedure of selecting a feature to split a node is summarized in Algorithm 2. As the temporal context only includes two features, we control the random selection of temporal features separately by introducing a prior probability $\tau \in (0, 1)$. In steps 8–11 of Algorithm 2, a random number θ and the predefined τ co-determine whether to add the temporal features to the feature candidate set.

3.4 Final classifier

With the mid-level feature $f(v)$, random forest is utilized for a second time to train our final classifier. We advocate the use of random forest because the classification technique is able to learn multiple classes discriminatively.

Given the mid-level feature $f(v)$ of a sub-volume v, each tree in the random forest gives a distribution of the posterior probabilities, and the sub-volume v is assigned to a particular label $\widetilde{y}(v)$ according to the average of the posterior probabilities of all trees (see (2)). Suppose that an original action video is segmented into S sub-volumes: $\{v_s | s = 1 : S\}$, we assign a label $\widetilde{y}(v_s)$ to each sub-volume v_s and classify the action video by majority voting over the labels of all sub-volumes.

4 Experiments

4.1 Human action datasets

The Weizmann dataset [25] is composed of 90 videos acted by 9 different people, each performing 10 actions: bend, jumping-jack (jack), jump-forward-on-two-legs (jump), jumping-in-place-on-two-legs (pjump), run, gallop-sideways (side), skip, walk, and wave-one-hands (wave1), wave-two-hands (wave2). We employ leave-one-subject-out cross-validation scheme and report the average accuracy. There are 10 runs in total, and for each run, videos of one actor are used for testing.

The UCF sports action dataset [26] contains 149 broadcast sports sequences from network news videos of ten different human actions: dive, golf swing (GSwing), swing at the high bar (HSwing), kick, weight-lift, horse-ride, run, skate, swing, and walk. Complex sports actions, drastic camera motion and large variation of human appearance augment the difficulty of classification. To increase the number of data samples, we follow [6] to extend the dataset by adding a horizontally flipped version of each sequence to the dataset. Two evaluation schemes are employed on the UCF dataset. Firstly, we randomly split the total 149 pairs of videos into a training set with 127 pairs and a testing set with 22 pairs for seven times, and report the average results. Secondly, we do leave-one-out testing for 20 randomly selected pairs of videos, and also report the average accuracy.

The multi-view IXMAS dataset [27] includes 11 actions, each of which is executed three times by 10 actors and recorded with 5 cameras simultaneously from different perspectives. These actions are: check watch, cross arms, scratch head, sit down, get up, turn around, walk, wave, punch, kick, point, and pick up. Considering the computation complexity, we only use one sequence of each action performed by each actor, and the leave-one-subject-out cross-validation scheme is employed on this dataset.

The Ballet dataset [11] consists of 44 sequences collected from an instructional Ballet DVD. Each frame of the sequences is labeled with 8 different actions: L-R present, R-L present, present, leg swing, jump, turn, hop, and stand still. We perform leave-one-out cross validation on this dataset, and report the per-frame classification rate since a sequence contains multiple actions and we cannot do per-video classification. Figure 2 shows some examples of the four datasets.

4.2 Experimental setting

The input of our approach is human-centered sub-volumes, so we first need to get the human-centered images. The extracted bounding boxes of human body are available in the UCF and Ballet datasets.

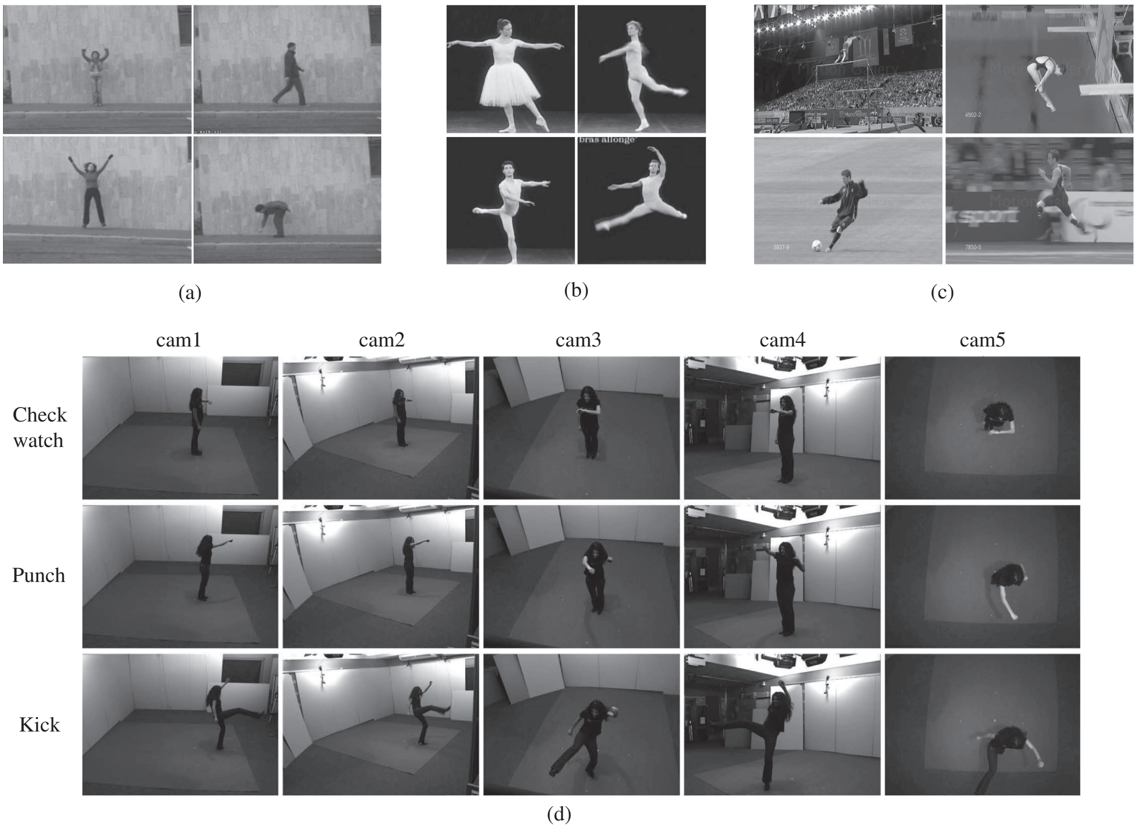


Figure 2 Examples of four datasets. (a) The Weizmann dataset; (b) the Ballet dataset; (c) the UCF dataset; (d) the IXMAS dataset.

For both Weizmann and IXMAS datasets, we fit a bounding box around the background subtracted silhouette. The bounding boxes are scaled to a fixed size (i.e., 50×50 pixels for the Ballet dataset and 80×40 pixels for the other datasets), and then 10 successive bounding boxes are concatenated to produce a sub-volume. Cuboids are extracted from the sub-volumes at regular positions and scales in space and time. We set the size of cuboids to $15 \times 15 \times 10$ and the stride between two adjacent cuboids to 5 pixels.

The key parameters of our model are the maximal depth of the tree $dep_{\max}$, the minimal number of data for a parent node $par_{\min}$, and the number of trees in a random forest $ntree$. For traditional random forest [22], each tree grows to maximal size without any limits. In consideration of the computation cost, we set $par_{\min}$ to 1 in the experiment, and tune the parameters $dep_{\max}$ and $ntree$ to control the growth of trees. Due to the randomness of random forest, we run the final classifier for 30 times with a group of fixed parameters, and report the average results.

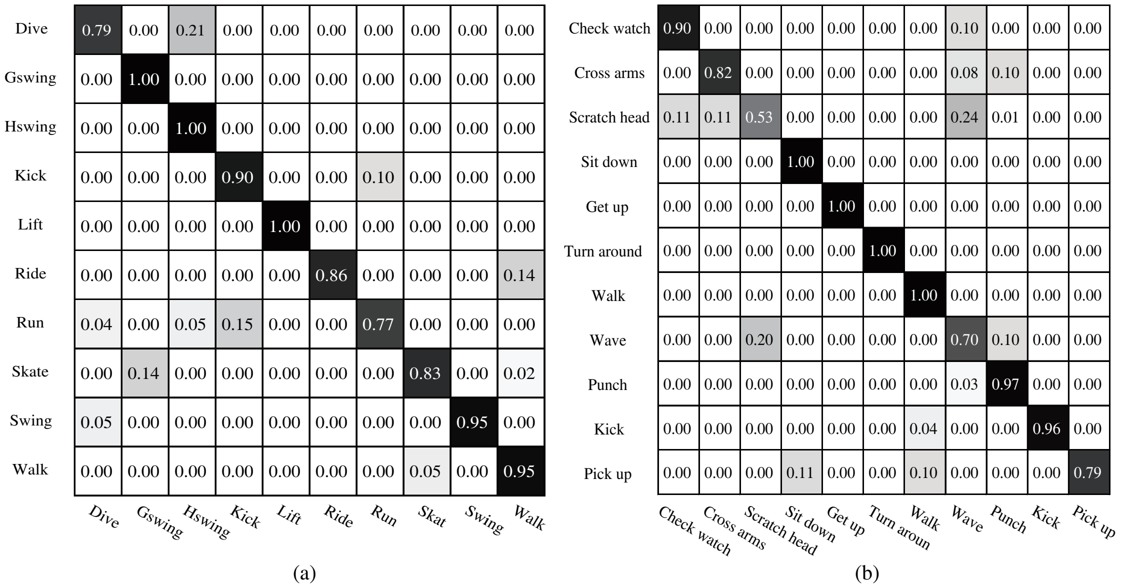
4.3 Experiment results

4.3.1 *Results on the Weizmann and UCF datasets*

Table 1 lists the action recognition results of our method compared with the previous work on the Weizmann dataset and the UCF dataset. Our method achieves better result than [28,29] and comparable result with [11] on the Weizmann dataset, which demonstrates the efficacy of our method. On the UCF dataset, all methods use different testing schemes, Ref. [29] adopts a five-fold cross-validation while [6,30,31] employ the leave-one-out cross-validation scheme. We evaluate our method with leave-one-out cross-validation and random split testing schemes, and results of the two schemes are better than those of the previous work. The methods of [6] and [30] are both based on bag-of-words model. Ref. [6] uses bag-of-words model to encode interest points and Ref. [30] extracts dense trajectories as low-level features. But no spatial-temporal context is employed in the two methods. In [31], a hierarchy of

Table 1 Recognition accuracy on the Weizmann and UCF datasets

Method	Weizmann	UCF (split)	UCF (leave-one-out)
Fathi and Mori [11] (2008)	100%	–	–
Wang et al. [6] (2009)	–	–	85.60%
Yao et al. [29] (2010)	97.80%	86.60%	–
Kovashka et al. [31] (2010)	–	–	87.27%
Wu et al. [28] (2010)	98.9%	–	–
Wang et al. [30] (2011)	–	–	88.2%
Our method	**100%**	**90.48%**	**95.50%**



	Dive	Gswing	Hswing	Kick	Lift	Ride	Run	Skat	Swing	Walk
Dive	0.79	0.00	0.21	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Gswing	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Hswing	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Kick	0.00	0.00	0.00	0.90	0.00	0.00	0.10	0.00	0.00	0.00
Lift	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00
Ride	0.00	0.00	0.00	0.00	0.00	0.86	0.00	0.00	0.00	0.14
Run	0.04	0.00	0.05	0.15	0.00	0.00	0.77	0.00	0.00	0.00
Skate	0.00	0.14	0.00	0.00	0.00	0.00	0.00	0.83	0.00	0.02
Swing	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.95	0.00
Walk	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.00	0.95

(a)

	Check watch	Cross arms	Scratch head	Sit down	Get up	Turn aroun	Walk	Wave	Punch	Kick	Pick up
Check watch	0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.10	0.00	0.00	0.00
Cross arms	0.00	0.82	0.00	0.00	0.00	0.00	0.00	0.08	0.10	0.00	0.00
Scratch head	0.11	0.11	0.53	0.00	0.00	0.00	0.00	0.24	0.01	0.00	0.00
Sit down	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Get up	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
Turn around	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00
Walk	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
Wave	0.00	0.00	0.20	0.00	0.00	0.00	0.00	0.70	0.10	0.00	0.00
Punch	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.97	0.00	0.00
Kick	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.96	0.00
Pick up	0.00	0.00	0.00	0.11	0.00	0.00	0.10	0.00	0.00	0.00	0.79

(b)

Figure 3 Confusion tables of our method on the UCF and IXMAS datasets. (a) The UCF dataset; (b) the IXMAS dataset.

discriminative space-time neighborhood feature is learned from low-level local features, nevertheless, their learning framework just models the connection of low-level features without introducing any discriminative information. In [29], a one-stage random forest method is utilized for action classification and detection; however our two-stage method achieves better results than [29] by employing high-level information to construct a discriminative mid-level feature. The confusion table of recognition results is depicted in Figigure 3(a). The recognition results are promising for the most actions. Moreover, some actions have very similar motion. For example, actors rotate fast in "dive", "Hswing" (swing at high bar) and "swing", which results in mis-recognition.

4.3.2 *Results on multi-view IXMAS dataset*

In Table 2, we summarize our results and compare them against other approaches on both single-view and multi-view recognitions of the IXMAS dataset. Our method outperforms [7,32,33] using mid-level feature learned from a single view. Bag-of-words based approaches [7,33] employ the context information by capturing the spatio-temporal relationships between interest points or building the connections between visual words; however, less high-level information with discrimination power is introduced in these two methods. In [32], a self-similarity matrix is utilized for action representation, but discarding all absolute view information reduces the discriminative power of this method. For multi-view recognition, we adopt a fusion strategy which averages the mid-level features of all views to get an integrated mid-level descriptor for classification. We achieve comparable results with [34] for single view recognition and our multi-view recognition results are significantly better than those of [34], which demonstrates the effectiveness of our

Table 2 Recognition accuracy on the IXMAS dataset

Method	cam1	cam2	cam3	cam4	cam5	all cams
Junejo et al. [32] (2008)	76.4%	77.6%	73.6%	68.8%	66.1%	72.7%
Weinland et al. [34] (2010)	85.8%	86.4%	88.0%	88.2%	74.7%	83.5%
Wu et al. [7] (2011)	81.9%	80.1%	77.1%	77.6%	73.4%	–
Liu et al. [33] (2011)	82.0%	81.0%	78.3%	82.4%	75.6%	–
Our method	**84.5%**	**84.7%**	**88.0%**	**82.9%**	**83.4%**	**88.0%**

Table 3 Recognition accuracy on the Ballet dataset

Method	L-R present	R-L present	Present	Leg swing	Jump	Turn	Hop	Still
Fathi and Mori[11]	72%	67%	51%	80%	9%	82%	24%	37%
Our method	84%	87%	70%	89%	10%	87%	41%	35%

Table 4 Comparison of our two-stage method with one-stage method on the Weizmann dataset

Method	**Overall**	Bend	Jack	Jump	Pjump	Run	Side	Skip	Walk	Wave1	Wave2
One-stage	**88.9%**	100%	100%	66.7%	100%	77.8%	77.8%	77.8%	100%	100%	88.9%
Two-stage	**100%**	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%

fusion strategy. Figure 3(b) illustrates the confusion table of recognition results when merging data obtained from multiple views.

4.3.3 *Results on the Ballet dataset*

We compare our results on the Ballet dataset with previous work [11], and list the recognition accuracy of each class in Table 3. We can see that our method performs better than [11] except for action ‘stand still’. A possible reason is that low-level features for this action is difficult to obtain.

4.4 Evaluations of the mid-level feature

The proposed method can be considered as a two-stage random forest framework, in which the first stage fuses multiple low-level features into the mid-level feature and the second stage takes the mid-level feature as the input of final classifier. To validate the effectiveness of the mid-level feature, we compare our two-stage method with the one-stage method which only uses low-level features. For one-stage method, the low-level features are directly inputted into a random forest classifier described in Subsection 3.4, and majority voting over the outputs of all cuboids makes the final classification. Table 4 lists the recognition accuracy of each class on the Weizmann dataset. As is shown in Table 4, our two-stage method achieves higher performance than the one-stage method, which demonstrates that the mid-level feature is discriminative for action recognition.

4.5 Evaluations of multiple low-level features fusion

The fusion of multiple low-level features (i.e., optical flow, HOG3D and temporal context) is evaluated on both UCF and IXMAS datasets. In Table 5, we summarize the results of fusing multiple low-level features, and compare them with that using a single low-level feature. As is shown in Table 5, it is interesting to observe that optical flow and HOG3D perform very differently on the UCF dataset and the IXMAS dataset. On the UCF dataset, the performance of HOG3D is slightly better than that of optical flow, but optical flow outperforms HOG3D on the IXMAS dataset. However, the proposed approach of fusing the above two low-level features achieves a better performance. We also compare the recognition rates of each class between different features in Figure 4. It is interesting to notice that, optical flow does

Table 5 Evaluations of multiple low-level features fusion

Method	UCF	IXMAS(cam1)	IXMAS(cam2)	IXMAS(cam3)	IXMAS(cam4)	IXMAS(cam5)
HOD3D	79.81%	64.8%	67.5%	67.9%	63.4%	66.3%
Optical flow(OF)	76.61%	79.7%	82.4%	84.7%	80.8%	78.1%
HOD3D & OF	90.10%	82.8%	83.3%	86.5%	81.8%	80.6%
HOD3D & OF & Temporal	90.48%	84.5%	84.7%	88.0%	82.9%	83.4%

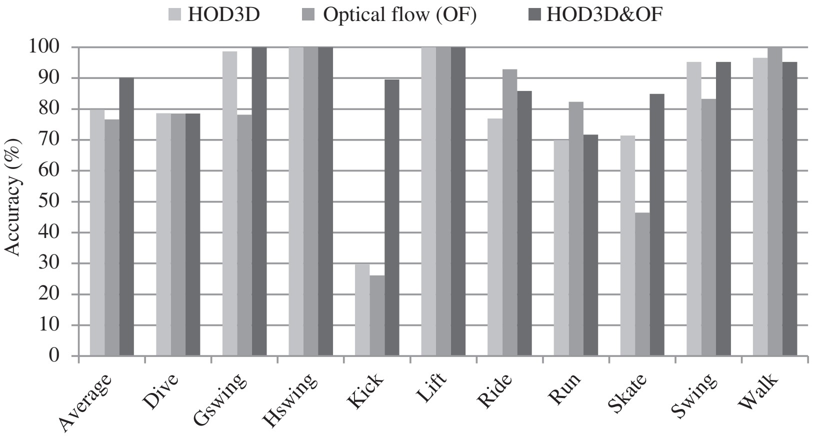


Figure 4 Comparison of low-level feature fusion and a single feature on the UCF dataset. We compare the recognition accuracy per class of multiple low-level features (violet) with that of HOG3D (orange) and optical flow (green).

well in classifying actions with drastic motion such as "run" while HOG3D is good at distinguishing actions by appearance such as "swing". Taking the actions of "kick" and "walk" for example, it is difficult to distinguish them due to the small and similar motions of leg; thereby algorithms of only using optical flow or HOG3D cannot work well. However, fusing the two types of low-level features, we can significantly improve the performance.

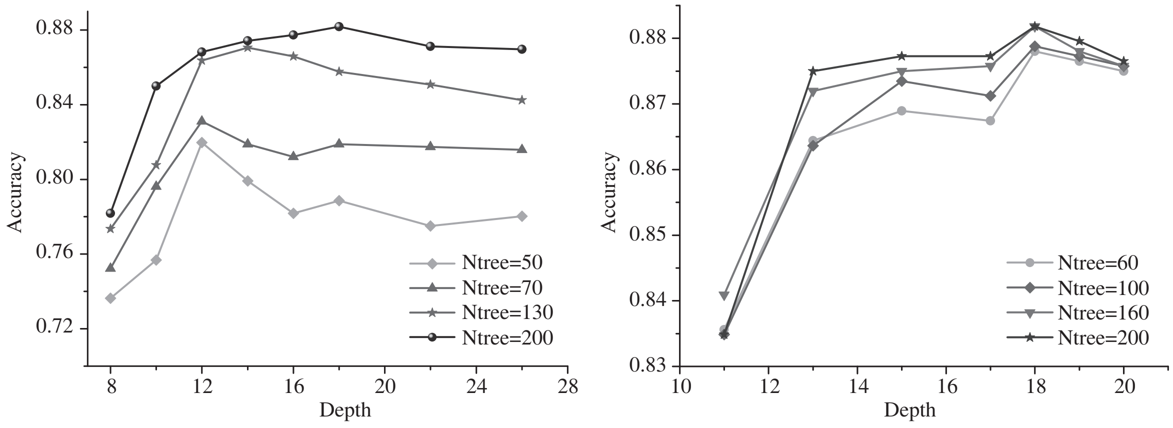
From Table 5, we can also observe that introducing temporal context leads to an improvement of recognition accuracy, especially for the IXMAS dataset. As actions in the IXMAS dataset are not so drastic as those in the UCF dataset, the information gain of optical flow and HOG3D is not very high when splitting a node in the construction of a tree. Therefore, for the IXMAS dataset, the temporal context is competitive in the selection of split feature and makes much contribution to the mid-level feature. That is why the IXMAS dataset gets a large improvement over the UCF dataset by using the temporal context. Additionally, we only employ a two-dimensional vector to characterize the temporal context, and a richer temporal feature generated by more functions of t/T could improve the performance.

4.6 Effects of varying parameters

We investigate the effect of several parameters of random forest on the UCF dataset. Random forest is adopted twice in our method, once for the construction of mid-level feature, and once for the final classification.

Figure 5(a) depicts how the parameters of random forest used to build the mid-level feature affect the performance. As is shown in Figure 5(a), the curves are first increasing and then decreasing due to the overfitting problem. Actually, the depth from which the forest is overfitting depends on the number of trees. For the curves with a larger number of trees, the maximum accuracy is obtained at deeper depth. Furthermore, it is interesting to observe that more trees can improve the performance.

In Figure 5(b) we show the quality of the final random forest classifier with different numbers of trees and different depths. We can observe that overfitting begins around depth 18 for all curves. Moreover, it is obvious that increasing the number of trees gradually improves the performance of the classifier.



(a) (b)

Figure 5 Training parameters (depth of trees and number of trees) vs. classification accuracy. (a) Training parameters of the random forest utilized to build the mid-level feature; (b) training parameters of the final random forest classifier.

5 Conclusion

This paper has presented a mid-level feature built upon the low-level optical flow and HOG3D of 3D cuboids as well as the temporal context between cuboids for human action recognition. An algorithm based on random forest is employed to effectively fuse the multiple low-level features. Evaluations on the Weizmann, UCF, Ballet and IXMAS datasets prove that our method performs robustly to variations of actions, and achieves high recognition accuracy. The proposed framework is general in nature and can be used with any type of low-level features. In future work, we plan to extend our framework to view-invariant action recognition. Moreover, as the temporal context is represented by a two-dimensional vector, which limits the contribution of temporal context to the mid-level feature, we should investigate how to exploit more functions of t/T to get a richer temporal context.

Acknowledgements

This work was supported in part by National Natural Science Foundation of China (Grant Nos. 61203274, 61203291) and Specialized Fund for Joint Building Program of Beijing municipal Education Commission.

References

1 Efros A A, Berg A C, Mori G, et al. Recognizing action at a distance. In: Proceedings of 9th IEEE Conference on Computer Vision (ICCV), Nice, 2003. 726–733

2 Thurau C, Hlavac V. Pose primitive based human action recognition in videos or still images. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, 2008. 1–8

3 Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Diego, 2005. 886–893

4 Laptev I, Marszalek M, Schmid C, et al. Learning realistic human actions from movies. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, 2008. 1–8

5 Klaser A, Marszalek M, Schmid C. A spatio-temporal descriptor based on 3D-gradients. In: Proceedings of the British Machine Vision Conference (BMVC), Leeds, 2008. 1–10

6 Wang H, Ullah M M, Klaser A, et al. Evaluation of local spatio-temporal features for action recognition. In: Proceedings of the British Machine Vision Conference (BMVC), London, 2009. 1–11

7 Wu X X, Xu D, Duan L X, et al. Action recognition using context and appearance distribution features. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, 2011. 489–496

8 Liu J G, Ali S, Shah M. Recognizing human actions using multiple features. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, 2008. 1–8

9 Wang Y, Mori G. Max-margin hidden conditional random fields for human action recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, 2009. 872–879

10 Han L, Wu X X, Liang W, et al. Discriminative human action recognition in the learned hierarchical manifold space. Image Vis Comput, 2010, 28: 836–849

11 Fathi A, Mori G. Action recognition by learning mid-level motion features. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, 2008. 1–8

12 Niebles J C, Li F F. A hierarchical model of shape and appearance for human action classification. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Minneapolis, 2007. 1–8

13 Kong Y, Zhang X Q, Hu W M, et al. Adaptive learning codebook for action recognition. Pattern Recogn Lett, 2011, 32: 1178–1186

14 Lu Z W, Peng Y X, Ip H H S. Spectral learning of latent semantics for action recognition. In: Proceedings of IEEE Conference on Computer Vision (ICCV), Barcelona, 2011. 1503–1510

15 Wang Y, Mori G. Hidden part models for human action recognition: probabilistic versus max-margin. IEEE Trans Pattern Anal Mach Intell, 2011, 33: 1310–1323

16 Niebles J C, Chen C W, Li F F. Modeling temporal structure of decomposable motion segments for activity classification. In: Proceedings of the 11th European Conference on Computer Vision (ECCV), Heraklion, 2010. 392–405

17 Raptis M, Kokkinos I, Soatto S. Discovering discriminative action parts from mid-level video representations. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, 2012. 1242–1249

18 Liu J G, Kuipers B, Savarese S. Recognizing human actions by attributes. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, 2011. 3337–3344

19 Bosch A, Zisserman A, Muoz X. Image classification using random forests and ferns. In: Proceedings of IEEE Conference on Computer Vision (ICCV), Rio de Janeiro, 2007. 1–8

20 Yu G, Yuan J S, Liu Z C. Unsupervised random forest indexing for fast action search. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, 2011. 865–872

21 Shotton J, Fitzgibbon A, Cook M, et al. Real-time human pose recognition in parts from single depth images. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), New York, 2011. 116–124

22 Breiman L. Random forests. Mach Learn, 2001, 45: 5–32

23 Lepetit V, Fua P. Keypoint recognition using randomized trees. IEEE Trans Pattern Anal Mach Intell, 2006, 28: 1465–1479

24 Breiman L. Randomizing outputs to increase prediction accuracy. Mach Learn, 2000, 40: 229–242

25 Blank M, Gorelick L, Shechtman E, et al. Actions as space-time shapes. In: Proceedings of 10th IEEE Conference on Computer Vision (ICCV), Beijing, 2005. 1395–1402

26 Rodriguez M D, Ahmed J, Shah M. Action mach: a spatio-temporal maximum average correlation height filter for action recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, 2008. 1–8

27 Weinland D, Boyer E, Ronfard R. Action recognition from arbitrary views using 3D exemplars. In: Proceedings of IEEE Conference on Computer Vision (ICCV), Rio de Janeiro, 2007. 1–7

28 Wu X X, Jia Y D, Liang W. Incremental discriminant-analysis of canonical correlations for action recognition. Pattern Recogn, 2010, 43: 4190–4197

29 Yao A, Gall J, Gool L V. A hough transform-based voting framework for action recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, 2010. 2061–2068

30 Wang H, Klaser A, Schmid C, et al. Action recognition by dense trajectories. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, 2011. 3169–3176

31 Kovashka A, Grauman K. Learning a hierarchy of discriminative space-time neighborhood features for human action recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, 2010. 2046–2053

32 Junejo I N, Dexter E, Laptev I, et al. Cross-view action recognition from temporal self-similarities. In: Proceedings of the 10th European Conference on Computer Vision (ECCV), Mardi, 2008. 1–19

33 Liu J G, Shah M, Kuipers B, et al. Cross-view action recognition via view knowledge transfer. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, 2011. 3209–3216

34 Weinland D, Ozuysal M, Fua P. Making action recognition robust to occlusions and viewpoint changes. In: Proceedings of the 11th European Conference on Computer Vision (ECCV), Heraklion, 2010. 635–648