Hindawi
Advances in Multimedia
Volume 2018, Article ID 3508350, 10 pages
https://doi.org/10.1155/2018/3508350

Research Article

Learning a Mid-Level Representation for Multiview Action Recognition

Cuiwei Liu,[1] **Zhaokui Li**,[1] **Xiangbin Shi**,[1,2] **and Chong Du**[3]

[1]*School of Computer Science, Shenyang Aerospace University, Shenyang City, Liaoning Province 110136, China*
[2]*School of Information, Liaoning University, Shenyang City, Liaoning Province 110136, China*
[3]*Shenyang Aircraft Design and Research Institute, Tawan Street 40, Shenyang City, Liaoning Province 110135, China*

Correspondence should be addressed to Cuiwei Liu; liucuiwei@sau.edu.cn

Received 13 November 2017; Revised 12 February 2018; Accepted 15 February 2018; Published 20 March 2018

Academic Editor: Martin Reisslein

Recognizing human actions in videos is an active topic with broad commercial potentials. Most of the existing action recognition methods are supposed to have the same camera view during both training and testing. And thus performances of these single-view approaches may be severely influenced by the camera movement and variation of viewpoints. In this paper, we address the above problem by utilizing videos simultaneously recorded from multiple views. To this end, we propose a learning framework based on multitask random forest to exploit a discriminative mid-level representation for videos from multiple cameras. In the first step, subvolumes of continuous human-centered figures are extracted from original videos. In the next step, spatiotemporal cuboids sampled from these subvolumes are characterized by multiple low-level descriptors. Then a set of multitask random forests are built upon multiview cuboids sampled at adjacent positions and construct an integrated mid-level representation for multiview subvolumes of one action. Finally, a random forest classifier is employed to predict the action category in terms of the learned representation. Experiments conducted on the multiview IXMAS action dataset illustrate that the proposed method can effectively recognize human actions depicted in multiview videos.

1. Introduction

Automatic recognition of human actions in videos becomes increasingly important in many applications such as intelligent video surveillance, smart home system, video annotation, and human-computer interaction. For example, finding out suspicious human behaviors in time is an essential task in intelligent video surveillance, and identifying fall actions of older people is of great importance for a smart home system. In recent years, a variety of action recognition approaches [1–5] have been proposed to solve single-view tasks, and some surveys [6–10] review the advances of single-view action recognition in detail. However, real-world videos bring about great challenges to single-view action recognition, since visual appearance of actions can be severely affected by viewpoint changes and self-occlusion.

Different from single-view approaches which utilize one camera to capture human actions, multiview action recognition methods exploit several cameras to record actions from multiple views and try to recognize actions by fusing multiview videos. One strategy is to handle the problem of multiview action recognition at classification level by annotating videos from multiple views separately and merging the predicted labels of all views. Pehlivan and Forsyth [11] designed a fusion scheme of videos from multiple views. They firstly annotated labels over frames and cameras using a nearest neighbor query technique and then employed a weighting scheme to fuse action judgments as the sequence label. Another group of methods resort to merging data from multiple views at feature level. These methods [12–15] utilize 3D or 2D models to build a discriminative representation of an action based on videos from multiple views. In fact, how to represent an action video with expressive features plays an especially important role in both multiview and single-view action recognition. A video representation with strong discriminative and descriptive ability is able to express human action reasonably and supply sufficient information to action classifier, which will lead to an improvement in recognition performance.

This paper presents a multiview action recognition approach with a novel mid-level action representation. A learning framework based on multitask random forest is proposed to exploit a discriminative mid-level representation from low-level descriptors of multiview videos. The input of our method is multiview subvolumes, each of which includes continuous human-centered figures. These subvolumes simultaneously record one action from different perspectives. And then we sample spatiotemporal cuboids from subvolumes at regular positions and extract multiple low-level descriptors to characterize each cuboid. During training, cuboids from multiple views sampled at four adjacent positions are grouped together to construct a multitask random forest by using action category and position as two related tasks, and a set of multitask random forests are constructed in this way. In testing, each cuboid is classified by the corresponding random forest, and a fusion strategy is employed to create an integrated histogram for describing cuboids sampled at a certain position of multiview subvolumes. Histograms of different positions are concatenated to a mid-level representation for subvolumes simultaneously recorded from multiple views. Moreover, the integrated histogram of multiview cuboids is created in terms of the distributions of both action categories and cuboid positions, which endows the learned mid-level representation with the ability of exploiting spatial context of cuboids. To achieve multiview action recognition, a random forest classifier is adopted to predict the category of this action. Figure 1 depicts the overview of our multitask random forest learning framework.

The remainder of this paper is organized as follows. After a brief overview of the related work in Section 2, we detailedly describe our method in Sections 3, 4, and 5. Then a description of experimental evaluation procedure followed by the analysis of results is given in Section 6. Finally, the paper concludes with discussions and conclusions in Section 7.

2. Related Work

The existing multiview action recognition methods fusing data at feature level can be roughly categorized into two groups, 3D based approaches and 2D based approaches.

Some action recognition methods based on 3D models have shown good performance on several public multiview action datasets. Weinland et al. [12] built 3D action representations based on invariant Fourier analysis of motion history volumes by using multiple view reconstructions. For the purpose of considering view dependency among cameras and adding full flexibility in camera configurations, they designed an exemplar-based hidden Markov model to characterize actions with 3D occupancy grids constructed from multiple views in another work [16]. Holte et al. [14] combined 3D optical flow of each view into enhanced 3D motion vector fields, which are described with the 3D Motion Context and the view-invariant Harmonic Motion Context in a view-invariant manner. Generally, 3D reconstruction from multiple cameras requires additional processing such as camera calibration, which would lead to high computational cost and reduce the flexibility. In order to overcome the limitation of 3D reconstruction from 2D images, some methods employ depth sensors for multiview action recognition. Hsu et al. [17] addressed the problem of view changes by using RGB-D cameras such as Microsoft Kinect. They constructed a view-invariant representation based on the Spatiotemporal Matrix and integrated the depth information into the spatiotemporal feature to improve the performance.

In recent years, different methods based on 2D models have been proposed for multiview action recognition. These methods aim to construct discriminative and view-invariant action representations from one or more descriptors. Souvenir and Babbs [13] learned low-dimensional and view-independent representations of actions recorded from multiple views by using manifold learning. In the work of [18], scale and location invariant features are calculated from human silhouettes to obtain sequences of multiview key poses, and action recognition is achieved through Dynamic Time Warping. Kushwaha et al. [19] extracted scale invariant contour-based pose features and uniform rotation invariant local binary patterns for view-invariant action recognition. Sargano et al. [20] learned discriminative and view-invariant descriptors for real-time multiview action recognition by using region-based geometrical and Hu-moments features extracted from human silhouettes. Chun and Lee [15] extracted local flow motion from multiview image sequences and estimated the dominant angle and intensity of optical flow for head direction identification. Then they utilized histogram of the dominant angle and intensity to represent each sequence and concatenated histograms of all views as the final feature of multiview sequences. Murtaza et al. [21] developed a silhouette-based view-independent action recognition scheme. They computed Motion History Images (MHI) for each view and employed Histograms of Oriented Gradients (HOG) to extract low-dimensional description of them. Gao et al. [22] evaluated seven popular regularized multitask learning algorithms on multiview action datasets and treated different actions as different tasks. In their work, videos from each view are handled separately. Hao et al. [23] employed a sparse coding algorithm to transfer the low-level features of multiple views into a discriminative and high-level semantics space and achieved action recognition by a multitask learning approach in which each action is considered as an individual task.

Besides, some other methods employ deep learning technique to learn discriminative features for multiview action recognition, and several neural networks are developed to build these deep-learned features directly from the raw data. Lei et al. [24] utilized convolutional neural network to extract effective and robust action features for continuous action segmentation and recognition under multiview setup.

The proposed method is also relevant to our previous work [25], in which a random forest based learning framework is designed for building mid-level representations of action videos. Different from [25], the proposed method aims to solve the problem of multiview action recognition, and an integrated mid-level representation is learned for an action depicted in videos recorded from multiple views. Meanwhile, our multitask random forest learning framework is able to effectively exploit the spatial context of cuboids.

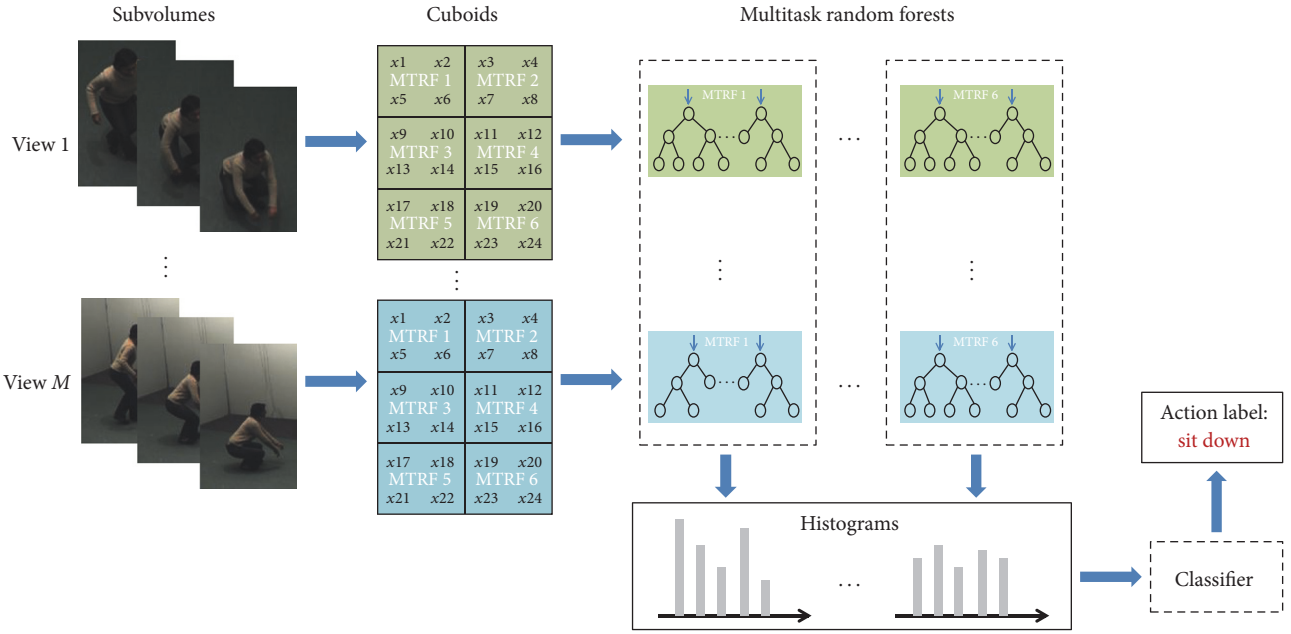

FIGURE 1: Framework of our method. Firstly, we densely sample spatiotemporal cuboids from subvolumes of M views. Suppose that 24 cuboids are extracted from one subvolume, and then six multitask random forests are constructed, each of which is built upon cuboids from multiple views sampled at four adjacent positions. Then all cuboids are classified by their corresponding random forests, and an integrated histogram is created to represent cuboids of all the M views sampled at the same position. The concatenation of histograms for all positions constitutes a mid-level representation of the input M subvolumes. At last, random forest is utilized as the final action classifier.

3. Overview of Our Method

Our goal is to recognize a human action by using videos recorded from multiple views. To this end, we propose a novel multitask random forest framework to learn a uniform mid-level feature for an action. In order to remove the influence of the background, we firstly employ a human body detector or tracker to obtain the human-centered figures from a video, and then a video is divided to a series of subvolumes with fixed size, each of which is a sequence of human-centered figures. We densely extract spatiotemporal cuboids (e.g., $15 \times 15 \times 10$) from subvolumes, and each of them is represented by multiple low-level features.

Our multitask random forest framework utilizes a fusion strategy to get an integrated histogram feature for cuboids sampled at the same position of subvolumes that simultaneously record an action from different views. Concretely, a multitask random forest is built upon cuboids extracted at four adjacent positions of multiview subvolumes, and thus we can construct a set of multitask random forests corresponding to different groups of positions. For the purpose of exploiting spatial context of cuboids, position of cuboid is treated as another task besides action category in the construction of multitask random forest. Decision trees in a multitask random forest vote on the action category and position of cuboids and generate a single histogram for cuboids sampled at the same position of simultaneously recorded multiview subvolumes, according to the distribution of both action category and cuboid position. The concatenation of histograms of all positions is normalized to get the mid-level representation for multiview subvolumes. For multiview action recognition, a random forest classifier is adopted to predict the category of this action.

4. Low-Level Features

Our multitask random forest based framework is general for merging multiple low-level features. In our implementation, we extract three complementary low-level features to describe the motion, appearance, and temporal context of the interested human. The optical flow feature computed from the entire human figure is able to characterize global motion information, the HOG3D spatial-temporal descriptor extracted from a single cuboid captures the local motion and appearance information, and the temporal context feature expresses the relative temporal location of cuboids. Therefore, the mid-level feature built upon the above three types of low-level features is more robust to video variations such as global deformation, local partial occlusion, and diversity of movement speed.

4.1. Optical Flow. Optical flow [35] is used to calculate the motion between two adjacent image frames. This motion descriptor shows favorable performance with noise, so it can tolerate the jitter of human figures caused by human detector or tracker. Given a sequence of human-centered figures, pixel-wise optical flow feature is calculated at each frame using Lucas-Kanade algorithm [36]. The optical flow vector field F is split into two scalar fields corresponding to the horizontal and vertical components of the flow, F_x and F_y. Then F_x and F_y are half-wave rectified into two nonnegative

channels F_x^+, F_x^- and F_y^+, F_y^-, respectively; namely, $F_x = F_x^+ + F_x^-$ and $F_y = F_y^+ + F_y^-$. Each channel is blurred with Gaussian filter and normalized to obtain the final four sparse and nonnegative channels, $\widehat{F}b_x^+$, $\widehat{F}b_x^-$, $\widehat{F}b_y^+$, and $\widehat{F}b_y^-$, which constitute the motion descriptor of each frame.

4.2. HOG3D. HOG3D [37] is a local spatiotemporal descriptor based on histograms of oriented 3D spatiotemporal gradients. It is an extension of HOG descriptor [38] to the video. 3D gradients are calculated with arbitrary spatial and temporal scales, followed by the orientation quantization using regular polyhedrons. A local support region is divided into $C1 \times C1 \times C2$ cells. An orientation histogram is computed for each cell, and the concatenation of all histograms is normalized to generate the final descriptor. In this paper, HOG3D descriptors are computed for cuboids densely sampled from human-centered subvolumes. We set $C1 = 4$ and $C2 = 3$, respectively, and utilize icosahedron with full orientation to quantize the 3D gradients of each cell. So the dimension of HOG3D feature is $4 \times 4 \times 3 \times 20 = 960$.

4.3. Temporal Context. Temporal context feature is characterized by the temporal relation among different cuboids and is regarded as a type of low-level feature in this paper. Given a video with L frames, a cuboid x is extracted from a subvolume which contains L_0 frames and begins with the lth frame. The temporal context of x is described as a two-dimensional vector $[l/L, |l/L| - 0.5]^T$, where l/L represents the temporal position of x in the whole video and $|l/L - 0.5|$ denotes the temporal offset of x relative to the center of the video.

5. Multitask Random Forest Learning Framework

We detail the proposed multitask random forest framework in this section. Suppose that an action is recorded by M cameras simultaneously, and we obtain V human-centered subvolumes with fixed size from each video, denoted as $\{\text{vol}^{m,v}\}_{m=1:M,v=1:V}$. Then we densely sample K spatiotemporal cuboids from every subvolume with particular size and stride and denote them by $\{x_k^{m,v}\}_{k=1:K,m=1:M,v=1:V}$, each of which is characterized by multiple low-level features. In order to exploit the spatial context of cuboids, we treat spatial position of cuboids as another type of annotations and employ cuboids of various action instances extracted at adjacent positions to build a multitask random forest by using both action labels and position labels. The proposed multitask random forest framework constructs an integrated histogram to describe cuboids $\{x_k^{m,v}\}_{m=1:M}$ sampled at position k of multiview subvolumes $\{\text{vol}^{m,v}\}_{m=1:M}$, and histograms of K cuboids are concatenated to create a unified mid-level representation for subvolumes $\{\text{vol}^{m,v}\}_{m=1:M}$ that are simultaneously recorded from multiple views.

5.1. Construction of Multitask Random Forest. Our training cuboids $\{x_{k,i}^{m,v}, y_i\}_{k=1:K,m=1:M,v=1:V_i,i=1:N}$ are extracted from subvolumes of N action instances, and each video of the ith instance generates V_i subvolumes. Cuboids of the ith instance share the same action label y_i, and the position label of cuboid $x_{k,i}^{m,v}$ is k. As is shown in Figure 1, we draw cuboids at regular positions of subvolumes and utilize training cuboids sampled at four adjacent positions to construct a multitask random forest. We totally obtain R random forests, denoted as $\{\text{MTRF}_r\}_{r=1:R}$.

Multitask random forest $\text{MTRF}_r = \{\text{Tree}_{r,t}^m\}_{m=1:M,t=1:T}$ is an ensemble of $M * T$ decision trees, and each tree only takes cuboids from a particular view as input. Decision trees $\{\text{Tree}_{r,t}^m\}_{t=1:T}$ of the same view share the original training set $D_r^m = \{x_{k,i}^{m,v}, y_i\}_{k\in\text{cub}(r),v=1:V_i,i=1:N}$, where $\text{cub}(r)$ represents the set of positions belonging to multitask random forest MTRF_r. As is shown in Figure 1, $\text{cub}(r)$ includes four positions that are adjacent in the direction of width or height. For example, cuboids at position 1, position 2, position 5, and position 6 (i.e., $x1$, $x2$, $x5$, and $x6$ in Figure 1) belong to MTRF_1, and thus $\text{cub}(1) = \{1, 2, 5, 6\}$.

In order to build decision tree $\text{Tree}_{r,t}^m$, we randomly sample about 2/3 of cuboids from the original training set D_r^m and obtain its own training dataset $D_{r,t}^m$, using bootstrap method. All of the training cuboids in $D_{r,t}^m$ go through the tree from the root. We split a node and the training cuboids assigned to it according to a particular feature chosen from a set of randomly sampled feature candidates. Since a cuboid is described by three types of low-level features (i.e., optical flow, HOG3D, and temporal context), two parameters $\gamma \in (0, 1)$ and $\tau \in (0, 1)$ are predefined to control the selection of feature candidates. Specifically, we generate two random numbers $\xi \in [0, 1]$ and $\theta \in [0, 1]$ to decide which type of low-level features is utilized for node split. If $\xi < \gamma$, then a quantity of optical flow features is randomly selected as feature candidates; otherwise some randomly selected HOG3D features comprise the set of feature candidates. Meanwhile, if $\theta < \tau$, then all temporal context features are added to the set of feature candidates. Each feature candidate divides the training cuboids at this node into two groups, and feature candidate with the largest information gain is chosen for node split. Then the node splits into two children nodes and each cuboid is sent to one of the children nodes. As the multitask random forest takes action category and cuboid position as two classification tasks, a random number $\rho \in [0, 1]$ and a prior probability $\mu \in (0, 1)$ codetermine which task is used to calculated the information gain of data split.

A node stops splitting when it has gotten to the limited tree depth $\text{dep}_{\max}$ or all samples arriving at this node belong to the same action category and position, and then it is regarded as a leaf. Two vectors $\mathbf{P} = [p^1, p^2, \ldots, p^A]$ and $\mathbf{Q} = [q^1, q^2, q^3, q^4]$ are created to store the distributions of action categories and cuboid positions, respectively. Here p^i denotes the posterior probability of cuboids arriving at the corresponding leaf node belonging to action i, and we have A actions in total. Similarly, q^i represents the proportion of cuboids at this leaf node being extracted from a particular position. Both $\mathbf{P}$ and $\mathbf{Q}$ of a leaf node are calculated from training cuboids assigned to it. The construction of a decision tree is summarized in Algorithm 1.

5.2. Construction of Mid-Level Features. During testing, our task is to recognize an action instance by using videos recorded from M views simultaneously. With respect to multiview subvolumes $\{\text{vol}^{m,v}\}_{m=1:M}$, cuboids $\{x_k^{m,v}\}_{m=1:M}$ sampled at position k are handled by the corresponding multitask random forest $\text{MTRF}_r = \{\text{Tree}_{r,t}^m\}_{m=1:M,t=1:T}$, satisfying the condition $k \in \text{cub}(r)$. Particularly, cuboid $x_k^{m,v}$ is dropped down decisions trees $\{\text{Tree}_{r,t}^m\}_{t=1:T}$; this means tree $\text{Tree}_{r,t}^m$ only takes the cuboid from view m as input. Suppose that the input cuboid $x_k^{m,v}$ arrives at leaf node $\vartheta(\text{Tree}_{r,t}^m, x_k^{m,v})$ of tree $\text{Tree}_{r,t}^m$, with histograms $\mathbf{P}_{\vartheta(\text{Tree}_{r,t}^m, x_k^{m,v})}$ and $\mathbf{Q}_{\vartheta(\text{Tree}_{r,t}^m, x_k^{m,v})}$ representing the distributions of action categories and cuboid positions. Then the average distributions voted by all decision trees can be calculated by

$$\begin{aligned}\overline{\mathbf{P}}\left(\{x_k^{m,v}\}_{m=1:M}\right) &= \frac{1}{T\cdot M}\sum_{m=1}^{M}\sum_{t=1}^{T}\mathbf{P}_{\vartheta(\text{Tree}_{r,t}^m, x_k^{m,v})},\\ \overline{\mathbf{Q}}\left(\{x_k^{m,v}\}_{m=1:M}\right) &= \frac{1}{T\cdot M}\sum_{m=1}^{M}\sum_{t=1}^{T}\mathbf{Q}_{\vartheta(\text{Tree}_{r,t}^m, x_k^{m,v})}.\end{aligned} \tag{1}$$

The concatenation of $\overline{\mathbf{P}}(\{x_k^{m,v}\}_{m=1:M})$ and $\overline{\mathbf{Q}}(\{x_k^{m,v}\}_{m=1:M})$ constitutes an integrated local descriptor of cuboids $\{x_k^{m,v}\}_{m=1:M}$, denoted by $\mathbf{h}(k,v) \in \mathbb{R}^{(A+4)\times 1}$. We deal with cuboids sampled at each sampling position separately and obtain a series of histograms $\{\mathbf{h}(k,v)\}_{k=1:K}$. Histograms of all the K positions are concatenated to a mid-level representation $\mathbf{f}(v)$ of subvolumes $\{\text{vol}^{m,v}\}_{m=1:M}$ that are simultaneously recorded from multiple perspectives.

$$\mathbf{f}(v) = [\mathbf{h}(1,v); \mathbf{h}(2,v); \ldots; \mathbf{h}(K,v)]. \tag{2}$$

Following [25], the out-of-bag estimate [39] is employed in the construction of mid-level representations during training to solve the overfitting problem. As described in Algorithm 1, decision trees $\{\text{Tree}_{r,t}^m\}_{t=1:T}$ of view m share an original training set $D_r^m = \{x_{k,i}^{m,v}, y_i\}_{k\in\text{cub}(r),v=1:V_i,i=1:N}$, and about 2/3 of cuboids in D_r^m constitute the bootstrap training set $D_{r,t}^m$ for tree $\text{Tree}_{r,t}^m$. The construction of local descriptor $\mathbf{h}(k,v,i)$ for training cuboids $\{x_{k,i}^{m,v}\}_{m=1:M}$ is the same as that for test cuboids, except that tree $\text{Tree}_{r,t}^m$ does not contribute to $\mathbf{h}(k,v,i)$ if it was trained on $x_{k,i}^{m,v}$. Accordingly, we rewrite (1) as

$$\begin{aligned}&\overline{\mathbf{P}}\left(\left\{x_{k,i}^{m,v}\right\}_{m=1:M}\right)\\ &\quad= \frac{\sum_{m=1}^{M}\sum_{t=1}^{T}\mathbf{P}_{\vartheta(\text{Tree}_{r,t}^m, x_{k,i}^{m,v})}\cdot\ell\left(\text{Tree}_{r,t}^m, x_{k,i}^{m,v}\right)}{\sum_{m=1}^{M}\sum_{t=1}^{T}\ell\left(\text{Tree}_{r,t}^m, x_{k,i}^{m,v}\right)},\\ &\overline{\mathbf{Q}}\left(\left\{x_{k,i}^{m,v}\right\}_{m=1:M}\right)\\ &\quad= \frac{\sum_{m=1}^{M}\sum_{t=1}^{T}\mathbf{Q}_{\vartheta(\text{Tree}_{r,t}^m, x_{k,i}^{m,v})}\cdot\ell\left(\text{Tree}_{r,t}^m, x_{k,i}^{m,v}\right)}{\sum_{m=1}^{M}\sum_{t=1}^{T}\ell\left(\text{Tree}_{r,t}^m, x_{k,i}^{m,v}\right)},\end{aligned} \tag{3}$$

where $\ell(r,t,m)$ is an indicator function defined by

$$\ell\left(\text{Tree}_{r,t}^m, x_{k,i}^{m,v}\right) = \begin{cases}1, & \text{if } \text{Tree}_{r,t}^m \text{ was not trained on } x_{k,i}^{m,v}\\ 0, & \text{if } \text{Tree}_{r,t}^m \text{ was not trained on } x_{k,i}^{m,v}\end{cases}. \tag{4}$$

Similarly, the mid-level representation $\mathbf{f}(v,i)$ of training subvolumes $\{\text{vol}_i^{m,v}\}_{m=1:M}$ is created by concatenating local descriptors of all positions.

$$\mathbf{f}(v,i) = [\mathbf{h}(1,v,i); \mathbf{h}(2,v,i); \ldots; \mathbf{h}(K,v,i)]. \tag{5}$$

5.3. Action Recognition with Mid-Level Representations. Given mid-level representations $\{\mathbf{f}(v,i)\}_{i=1:N,v=1:V_i}$ of training subvolumes, where $\mathbf{f}(v,i)$ denotes the integrated feature of multiview subvolumes $\{\text{vol}_i^{m,v}\}_{m=1:M}$, we train a random forest classifier [39] which is able to learn multiple categories discriminatively.

For a new action instance $\{\mathbf{f}(v)\}_{v=1:V}$, all of the decision trees in random forest vote on the action category of each sample and assign a particular action label $y(v)$ to $\mathbf{f}(v)$. According to majority voting, we predict the final action category of this instance as

$$y^* = \arg\max_{y=1:A}\sum_{v=1:V}\mathbf{I}(y(v) = y), \tag{6}$$

where $\mathbf{I}(a=b)$ is an indicator function; that is, $\mathbf{I}(a=b)$ is 1 if $a=b$ and 0 otherwise.

6. Experiments

6.1. Human Action Datasets. Experiments are conducted on the multiview IXMAS action dataset [12] and the MuHAVi-MAS dataset [31] to evaluate the effectiveness of the proposed method.

The IXMAS Dataset. It consists of 11 actions performed by 10 actors, including "check watch", "cross arms", "scratch head", "sit down", "get up", "turn around", "walk", "wave", "punch", "kick", and "pick up". Five cameras simultaneously recorded these actions from different perspectives, that is, four side views and one top view. This dataset presents an increased challenge since actors can freely choose their position and orientation. Thus, there are large inter-view and intra-view viewpoint variations of human actions in this dataset, which make it widely used to evaluate the performance of multiview action recognition methods.

The MuHAVi-MAS Dataset. It contains 136 manually annotated silhouette sequences of 14 primitive actions: "CollapseLeft", "CollapseRight", "GuardToKick", "GuardToPunch", "KickRight", "PunchRight", "RunLeftToRight", "RunRightToLeft", "StandupLeft", "StandupRight", "TurnBackLeft", "TurnBackRight", "WalkLeftToRight", and "WalkRightToLeft". Each action is performed several times by 2 actors and captured by 2 cameras from different views.

6.2. Experimental Setting. Since our method takes human-centered subvolumes recorded from multiple views as input,

Input: The original training dataset $D_r^m = \{x_{k,i}^{m,v}, y_i\}_{k\in \mathrm{cub}(r), v=1:V_i, i=1:N}$;
Predefined parameters $\mathrm{dep}_{\max}$, $\gamma \in (0,1)$, $\tau \in (0,1)$, and $\mu \in (0,1)$;
Output: Decision tree $\mathrm{Tree}_{r,t}^m$;
(1) Build a bootstrap dataset $D_{r,t}^m$ by random sampling from D_r^m with replacement;
(2) Create a root node and set its depth to 1, then assign all cuboids in $D_{r,t}^m$ to it;
(3) Initialize an unsettled node queue $\Upsilon = \emptyset$ and push the root node into Υ;
(4) **while** $\Upsilon \neq \emptyset$ **do**
(5) Pop the first node ϑ in Υ;
(6) **if** depth of ϑ is larger than $\mathrm{dep}_{\max}$ **or** cuboids assigned to ϑ belong to the same action and position **then**
(7) Label node ϑ as a leaf, and then calculate **P** and **Q** from cuboids at node ϑ;
(8) Add a triple (ϑ, P_a, P_c) into decision tree $\mathrm{Tree}_{r,t}^m$;
(9) **else**
(10) Initialize the feature candidate set $\Delta = \emptyset$;
(11) **if** random number $\xi < \gamma$ **then**
(12) Add a set of randomly selected optical flow features to Δ;
(13) **else**
(14) Add a set of randomly selected HOG3D features to Δ;
(15) **end if**
(16) **if** random number $\theta < \tau$ **then**
(17) Add two-dimensional temporal context features to Δ;
(18) **end if**
(19) maxgain $= -\infty$, generate a random number ρ;
(20) **for** each $\delta_d \in \Delta$ **do**
(21) **if** $\rho < \mu$ **then**
(22) Search for the corresponding threshold ς_d and compute information gain $g(\delta_d)$ in terms of action labels of cuboids arriving at ϑ;
(23) **else**
(24) Search for the corresponding threshold ς_d and compute information gain $g(\delta_d)$ in terms of positions of cuboids arriving at ϑ;
(25) **end if**
(26) **if** $g(\delta_d) >$ maxgain **then**
(27) $\delta^* = \delta_d, \varsigma^* = \varsigma_d$;
(28) **end if**
(29) **end for**
(30) Create left children node ϑ_L and right children node ϑ_R, set their depth to dep + 1, and assign each cuboid arriving at ϑ to ϑ_L or ϑ_R according to δ^* and ς^*; then push node δ^* and ς^* into Υ;
(31) Add a quintuple $(\vartheta, \vartheta_L, \vartheta_R, \delta^*, \varsigma^*)$ into decision tree $\mathrm{Tree}_{r,t}^m$;
(32) **end if**
(33) **end while**
(34) **return** Decision tree $\mathrm{Tree}_{r,t}^m$;

Algorithm 1: Construction of a decision tree.

we utilize background subtraction technique to obtain human silhouettes and fit a bounding box around each silhouette. In our implementation, a subvolume is composed of 10 successive human-centered bounding boxes which are scaled to 80 × 40 pixels. We densely extract 84 cuboids from each subvolume with particular size (i.e., 15 × 15 × 10), and the strides between cuboids are 5 pixels.

6.3. Experimental Results. We compare our method with state-of-the-art methods on two datasets, and experimental results on the IXMAS dataset and the MuHAVi-MAS dataset are illustrated in Tables 1 and 2, respectively. In our experiments, the leave-one-actor-out cross-validation strategy is adopted on both datasets. We execute the random forest classifier for 10 times and report the recognition accuracy by averaging over the results of 10 classifiers.

6.3.1. Results on the IXMAS Dataset. As shown in Table 1, our method significantly outperforms all the recently proposed methods for multiview action recognition, which demonstrates the effectiveness of the proposed learning framework based on multitask random forest. The confusion matrix of multiview action recognition results is depicted in Figure 2. We can observe from Figure 2 that the proposed method achieves promising performance on most actions, among which four actions (i.e., "sit down", "get up", "walk", and "punch") are correctly recognized. Meanwhile, some actions have similar motion, which may result in misclassification. For example, it is difficult to distinguish actions "cross arms", "scratch head", and "wave", since they all involve motion of the upper limb. Similarly, actors crouch in both actions of "sit down" and action "pick up", which may be a possible reason for the misclassification of "pick up".

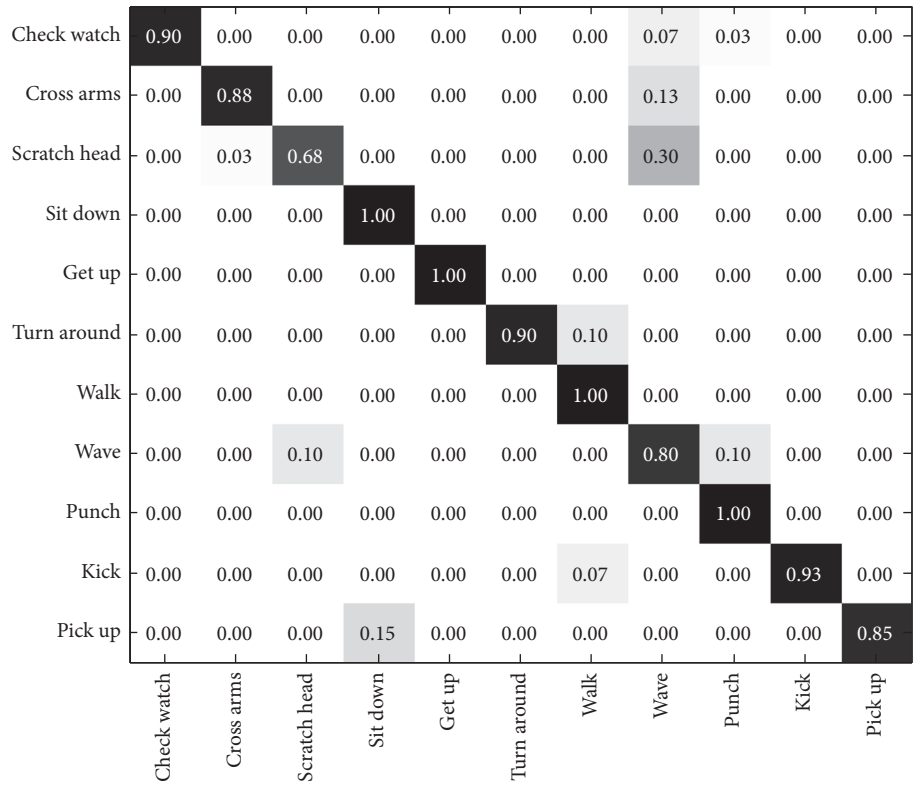

Figure 2: Confusion matrix of our multiview action recognition method on the IXMAS dataset.

Table 1: Action recognition accuracy comparison with state-of-the-art methods for both multiview action recognition and single-view action recognition on the IXMAS dataset.

Method	View 1	View 2	View 3	View 4	View 5	Average	Multiview
Junejo et al. [26]	76.4%	77.6%	73.6%	68.8%	66.1%	72.5%	72.7%
Weinland et al. [27]	85.8%	86.4%	88.0%	88.2%	74.7%	84.6%	83.5%
Wu et al. [28]	81.9%	80.1%	77.1%	77.6%	73.4%	78.0%	–
Chaaraoui et al. [18]	–	–	–	–	–	–	85.9%
Liu et al. [25]	84.5%	84.7%	88.0%	82.9%	83.4%	84.7%	88.0%
Pehlivan and Forsyth [11]	81.3%	86.3%	86.3%	85.9%	77.6%	83.5%	72.5%
Aryanfar et al. [29]	–	–	–	–	–	–	88.1%
Hsu et al. [17]	–	–	–	–	–	–	81.8%
Chun and Lee [15]	–	–	–	–	–	–	83.0%
Wang et al. [30]	–	–	–	–	–	–	84.9%
Sargano et al. [20]	–	–	–	–	–	–	89.7%
Our method	89.6%	88.4%	85.8%	78.2%	86.9%	85.8%	90.3%

The proposed method is also compared with other methods for single-view action recognition, and the results are summarized in Table 1. Videos of different views are handled separately, and a mid-level representation is created for a single video. Concretely, we build a set of multitask random forests for each view by using cuboids extracted from subvolumes of this view. Accordingly, all the decision trees in a multitask random forest share an original training dataset composed of cuboids from a certain view. It is observed that the proposed method performs better than [26, 28] on all of the five views. We can see that our method is competitive with [11, 25, 27] on two views and achieves much better performance on three views. However, our method is able to outperform the above three methods by fusing videos of multiple views into an integrated representation.

6.3.2. Results on the MuHAVi-MAS Dataset. As is shown in Table 2, our method achieves much better performance than the listed methods on the MuHAVi-MAS dataset. The promising results demonstrate the effectiveness of the proposed method. Figure 3 reveals the confusion matrix of our results. We can see that our method can correctly identify

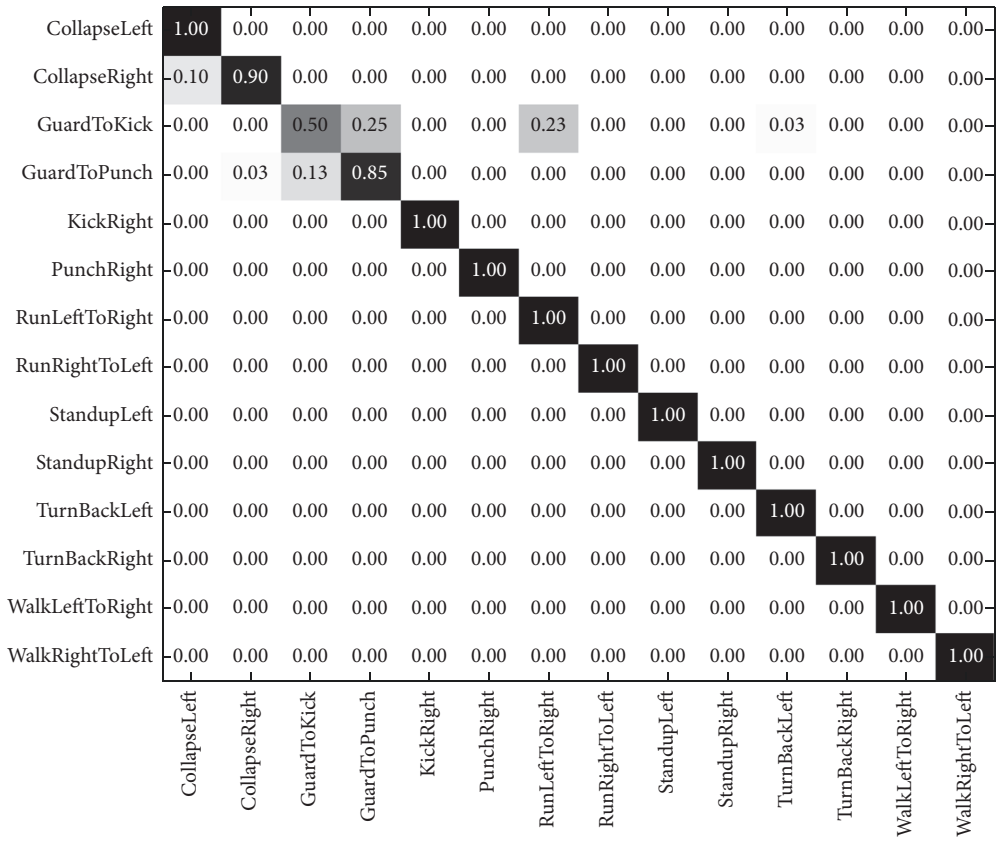

Figure 3: Confusion matrix of our action recognition method on the MuHAVi-MAS dataset.

eleven of the fourteen actions, that is, "CollapseLeft", "KickRight", "PunchRight", "RunLeftToRight", "RunRightToLeft", "StandupLeft", "StandupRight", "TurnBackLeft", "TurnBackRight", "WalkLeftToRight", and "WalkRightToLeft". It is also observable that our method does not do well in distinguishing "GuardToKick" and "GuardToPunch", since they have very similar motion.

6.4. Effects of Parameters. In this section, we evaluate the effect of two parameters in the construction of multitask random forest on the IXMAS dataset.

In consideration of the computation cost, we limit the depth of each decision tree to dep_{max}, and Figure 4 depicts the accuracy of both single-view action recognition on five perspectives and multiview action recognition with different tree depths. Generally, the curves first rise and then decline slightly with the increasement of tree depth. One possible reason is that large decision trees may overfit training data. Meanwhile, it is interesting to observe that the depths from which the forests are overfitting vary on different views. Furthermore, we can observe from Figure 4 that multiview action recognition method is able to achieve better performance than single-view action recognition with all tree depths, which demonstrates the effectiveness of the multiview fusion scheme.

Another key parameter of our method is the prior probability $\mu \in (0, 1)$. Our multitask random forest is designed

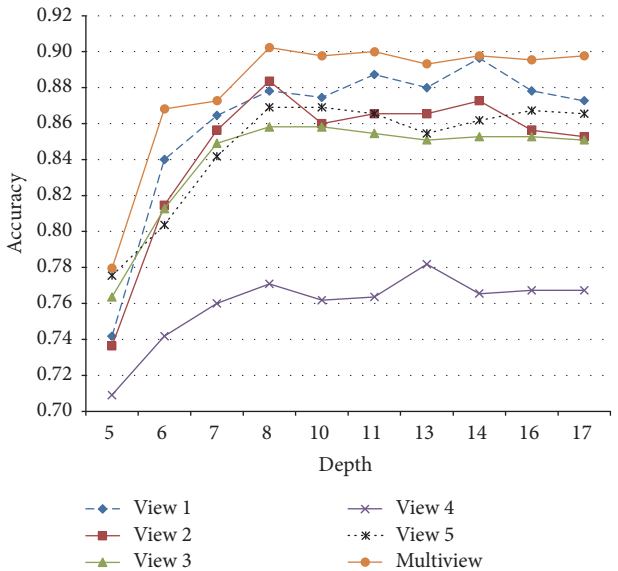

Figure 4: Action recognition accuracy with different tree depths on the IXMAS dataset.

to solve two tasks including action recognition and position classification. For the purpose of node split, we introduce the parameter μ to select a certain task for calculating the

Table 2: Action recognition accuracy of different methods on the MuHAVi-MAS dataset.

Method	Accuracy
Singh et al. [31]	61.8%
Orrite et al. [32]	75.0%
Cheema et al. [33]	75.5%
Murtaza et al. [34]	81.6%
Our method	91.2%

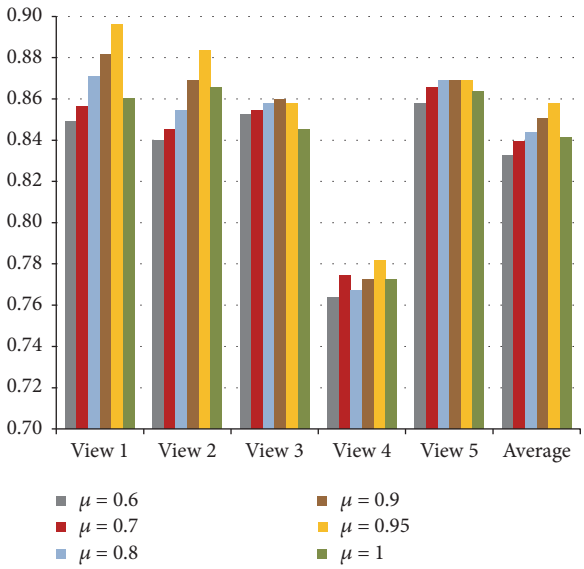

Figure 5: Action recognition accuracy of different values of μ on the IXMAS dataset.

information gain of data split. More concretely, μ denotes the probability that the action recognition task is selected at each node. We tune the value of μ to investigate how it affects the performance and summarize the action recognition results in Figure 5. It should be noted that multitask random forest is reduced to random forest which takes action recognition as its only task if μ is set to 1. From Figure 5 we can observe that action recognition results of multitask random forest (e.g., $\mu = 0.8, 0.9, 0.95$) are better than that of single-task random forest (i.e., $\mu = 1$), which demonstrates the effectiveness of our multitask random forest learning framework.

7. Conclusion

We presented a learning framework based on multitask random forest in order to exploit a discriminative mid-level representation for videos from multiple views. Our method starts from multiview subvolumes with fixed size, each of which is composed of continuous human-centered figures. Densely sampled spatiotemporal cuboids are extracted from subvolumes and three types of low-level descriptors are utilized to capture the motion, appearance, and temporal context of each cuboid. Then a multitask random forest is built upon cuboids from multiple views that are sampled at four adjacent positions, taking action category and position as two tasks. Each cuboid is classified by its corresponding random forest, and a fusion strategy is employed to create an integrated histogram for describing cuboids sampled at a certain position of multiview subvolumes. Concatenation of histograms for different positions is utilized as a mid-level representation for subvolumes simultaneously recorded from multiple views. Experiments on the IXMAS action dataset show that the proposed method is able to achieve promising performance.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported in part by the Natural Science Foundation of China (NSFC) under Grants no. 61602320 and no. 61170185, Liaoning Doctoral Startup Project under Grants no. 201601172 and no. 201601180, Foundation of Liaoning Educational Committee under Grants no. L201607, no. L2015403, and no. L2014070, and the Young Scholars Research Fund of SAU under Grant no. 15YB37.

References

[1] I. Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, no. 2-3, pp. 107–123, 2005.

[2] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '08)*, June 2008.

[3] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 3169–3176, June 2011.

[4] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles, "ActivityNet: A large-scale video benchmark for human activity understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 961–970, USA, June 2015.

[5] C. Liu, X. Wu, and Y. Jia, "A hierarchical video description for complex activity understanding," *International Journal of Computer Vision*, vol. 118, no. 2, pp. 240–255, 2016.

[6] R. Poppe, "Vision-based human motion analysis: an overview," *Computer Vision and Image Understanding*, vol. 108, no. 1-2, pp. 4–18, 2007.

[7] R. Poppe, "A survey on vision-based human action recognition," *Image and Vision Computing*, vol. 28, no. 6, pp. 976–990, 2010.

[8] D. Weinland, R. Ronfard, and E. Boyer, "A survey of vision-based methods for action representation, segmentation and recognition," *Computer Vision and Image Understanding*, vol. 115, no. 2, pp. 224–241, 2011.

[9] M. Vrigkas, C. Nikou, and I. A. Kakadiaris, "A review of human activity recognition methods," *Frontiers in Robotics and AI*, vol. 2, article 28, 2015.

[10] H. Xu, Q. Tian, Z. Wang, and J. Wu, "A survey on aggregating methods for action recognition with dense trajectories," *Multimedia Tools and Applications*, vol. 75, no. 10, pp. 5701–5717, 2016.

[11] S. Pehlivan and D. A. Forsyth, "Recognizing activities in multiple views with fusion of frame judgments," *Image and Vision Computing*, vol. 32, no. 4, pp. 237–249, 2014.

[12] D. Weinland, R. Ronfard, and E. Boyer, "Free viewpoint action recognition using motion history volumes," *Computer Vision and Image Understanding*, vol. 104, no. 2-3, pp. 249–257, 2006.

[13] R. Souvenir and J. Babbs, "Learning the viewpoint manifold for action recognition," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, USA, June 2008.

[14] M. B. Holte, T. B. Moeslund, N. Nikolaidis, and I. Pitas, "3D human action recognition for multi-view camera systems," in *Proceedings of the 2011 International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission, 3DIMPVT 2011*, pp. 342–349, China, May 2011.

[15] S. Chun and C.-S. Lee, "Human action recognition using histogram of motion intensity and direction from multiple views," *IET Computer Vision*, vol. 10, no. 4, pp. 250–256, 2016.

[16] D. Weinland, E. Boyer, and R. Ronfard, "Action recognition from arbitrary views using 3D exemplars," in *Proceedings of the IEEE 11th International Conference on Computer Vision (ICCV '07)*, pp. 1–7, October 2007.

[17] Y.-P. Hsu, C. Liu, T.-Y. Chen, and L.-C. Fu, "Online view-invariant human action recognition using rgb-d spatio-temporal matrix," *Pattern Recognition*, vol. 60, pp. 215–226, 2016.

[18] A. A. Chaaraoui, P. Climent-Pérez, and F. Flórez-Revuelta, "Silhouette-based human action recognition using sequences of key poses," *Pattern Recognition Letters*, vol. 34, no. 15, pp. 1799–1807, 2013.

[19] A. K. S. Kushwaha, S. Srivastava, and R. Srivastava, "Multi-view human activity recognition based on silhouette and uniform rotation invariant local binary patterns," *Multimedia Systems*, vol. 23, no. 4, pp. 451–467, 2017.

[20] A. B. Sargano, P. Angelov, and Z. Habib, "Human action recognition from multiple views based on view-invariant feature descriptor using support vector machines," *Applied Sciences (Switzerland)*, vol. 6, no. 10, article no. 309, 2016.

[21] F. Murtaza, M. H. Yousaf, and S. A. Velastin, "Multi-view human action recognition using 2D motion templates based on MHIs and their HOG description," *IET Computer Vision*, vol. 10, no. 7, pp. 758–767, 2016.

[22] Z. Gao, S. H. Li, G. T. Zhang, Y. J. Zhu, C. Wang, and H. Zhang, "Evaluation of regularized multi-task leaning algorithms for single/multi-view human action recognition," *Multimedia Tools and Applications*, vol. 76, no. 19, pp. 20125–20148, 2017.

[23] T. Hao, D. Wu, Q. Wang, and J.-S. Sun, "Multi-view representation learning for multi-view action recognition," *Journal of Visual Communication and Image Representation*, vol. 48, pp. 453–460, 2017.

[24] J. Lei, G. Li, J. Zhang, Q. Guo, and D. Tu, "Continuous action segmentation and recognition using hybrid convolutional neural network-hidden Markov model model," *IET Computer Vision*, vol. 10, no. 6, pp. 537–544, 2016.

[25] C. Liu, M. Pei, X. Wu, Y. Kong, and Y. Jia, "Learning a discriminative mid-level feature for action recognition," *Science China Information Sciences*, vol. 57, no. 5, pp. 1–13, 2014.

[26] I. N. Junejo, E. Dexter, I. Laptev, and P. Pérez, "Cross-View Action Recognition from Temporal Self-similarities," in *Computer Vision – ECCV 2008*, vol. 5303 of *Lecture Notes in Computer Science*, pp. 293–306, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[27] D. Weinland, M. Özuysal, and P. Fua, "Making action recognition robust to occlusions and viewpoint changes," in *European Conference on Computer Vision*, pp. 635–648, 2010.

[28] X. Wu, D. Xu, L. Duan, and J. Luo, "Action recognition using context and appearance distribution features," in *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011*, pp. 489–496, USA, June 2011.

[29] A. Aryanfar, R. Yaakob, A. A. Halin, M. N. Sulaiman, K. A. Kasmiran, and L. Mohammadpour, "Multi-view human action recognition using wavelet data reduction and multi-class classification," *Procedia Computer Science*, vol. 62, no. 27, pp. 585–592, 2015.

[30] W. Wang, Y. Yan, L. Zhang, R. Hong, and N. Sebe, "Collaborative Sparse Coding for Multiview Action Recognition," *IEEE MultiMedia*, vol. 23, no. 4, pp. 80–87, 2016.

[31] S. Singh, S. A. Velastin, and H. Ragheb, "MuHAVi: a multicamera human action video dataset for the evaluation of action recognition methods," in *Proceedings of the 7th IEEE International Conference on Advanced Video and Signal Based (AVSS '10)*, pp. 48–55, September 2010.

[32] C. Orrite, M. Rodriguez, E. Herrero, G. Rogez, and S. A. Velastin, "Automatic segmentation and recognition of human actions in monocular sequences," in *Proceedings of the 22nd International Conference on Pattern Recognition, ICPR 2014*, pp. 4218–4223, Sweden, August 2014.

[33] S. Cheema, A. Eweiwi, C. Thurau, and C. Bauckhage, "Action recognition by learning discriminative key poses," in *Proceedings of the Proceeding of the IEEE International Conference on Computer Vision Workshops (ICCV '11)*, pp. 1302–1309, Barcelona, Spain, November 2011.

[34] F. Murtaza, M. H. Yousaf, and S. A. Velastin, "Multi-view Human Action Recognition Using Histograms of Oriented Gradients (HOG) Description of Motion History Images (MHIs)," in *Proceedings of the 13th International Conference on Frontiers of Information Technology, FIT 2015*, pp. 297–302, Pakistan, December 2015.

[35] A. A. Efros, A. C. Berg, G. Mori, and J. Malik, "Recognizing action at a distance," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 726–733, Nice, France, October 2003.

[36] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the International Joint Conference on Artificial Intelligence*, vol. 81, pp. 674–679, 1981.

[37] A. Kläser, M. Marszałek, and C. Schmid, "A spatio-temporal descriptor based on 3D-gradients," in *Proceedings of the 19th British Machine Vision Conference (BMVC '08)*, pp. 995–1004, September 2008.

[38] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 1, pp. 886–893, June 2005.

[39] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.