Front. Electr. Electron. Eng. 2012, 7(2): 224–241
DOI 10.1007/s11460-011-0175-6

RESEARCH ARTICLE

Junxia GU, Xiaoqing DING, Shenjing WANG

Action recognition from arbitrary views using 3D-key-pose set

Abstract Recovering three-dimensional (3D) human pose sequence from arbitrary view is very difficult, due to loss of depth information and self-occlusion. In this paper, view-independent 3D-key-pose set is selected from 3D action samples, for the purpose of representing and recognizing those same actions from a single or few cameras without any restriction of the relative orientations between cameras and subjects. First, 3D-key-pose set is selected from the 3D human joint sequences of 3D training action samples that are built from multiple viewpoints. Second, 3D key pose sequence, which matches best with the observation sequence, is selected from the 3D-key-pose set to represent the observation sequence of arbitrary view. 3D key pose sequence contains many discriminative view-independent key poses but cannot accurately describe pose of every frame in the observation sequence. Considering the above reasons, pose and dynamic of action are modeled respectively in this paper. Exemplar-based embedding and probability of unique key pose are applied to model pose property. Complementary dynamic feature is extracted to model these actions that share the same poses but have different dynamic features. Finally, these action models are fused to recognize observation sequence from a single or few cameras. Effectiveness of the proposed approach is demonstrated with experiments on IXMAS dataset.

Keywords action representation, action recognition, 3D-key-pose set, 3D key pose sequence, action models fusion

1 Introduction

Video-based study of human action recognition has been receiving increased attention over the past decades, due to the requirement of intelligent video surveillance. Increasing awareness of security issues, motion analysis is becoming increasingly important in surveillance systems. Action recognition is an important requirement for understanding what the person is doing. Fixing the camera viewpoint and considering only one view of the action is not suitable for solving practical problems that may arise during a surveillance scenario where an action can be observed from different viewing angles [1]. Current intelligent surveillance systems are in urgent need of non-invasive and view-independent research on human action analysis.

Various views, self-occluding, and loss of depth information make video-based action recognition very challenging. The same pose can have many different observations, and the different poses can result in same observation [2]. Therefore, "view invariance" has become very important in action recognition. Several types of view-independent action recognition approaches have been brought forward. These approaches can be classified into monocular methods [1,3], multiview methods [4,5], and three-dimensional (3D) methods [6,7]. Monocular methods recognize action from monocular image sequences. They attempt to obtain some viewpoint-invariance pose quantities through several corresponding points. These corresponding points, such as human joints, are obtained manually or by a marker-based motion capture system. However, the corresponding points have to be obtained automatically in many applications. Multi-view methods learn action models in multiple viewpoints and fuse multiple models in the classification level. The difficulties of these methods lie in the fusion of multiple models. In the real world, subjects perform actions in 3D space, and actions are presented as the 3D pose sequences. However, cameras can only capture two-dimensional (2D) image sequence in the projection plane. In this paper, the image sequence captured by camera is called 2D image action sequence, and the action sequence in 3D space is called 3D action sequence. For 3D recognition methods, 3D reconstruction technique is used to obtain 3D action sequence. In 3D space, action sequences can be observed in the same manner as that in the real world, and the viewpoint difficulty and self-occluding problem can be

Received April 28, 2011; accepted June 27, 2011

Junxia GU, Xiaoqing DING (✉), Shenjing WANG
Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
E-mail: dingxq@tsinghua.edu.cn

easily solved. The difficult parts of 3D methods are hardware setting and robust 3D reconstruction. Multiple video streams need to be simultaneously captured from multiple static calibrated cameras. Then, 3D reconstruction technique is used to obtain volume data. However, it is difficult to obtain robust volume data in surveillance and content-analysis scenarios at present. A new kind of method [8–10] is proposed to avoid the 3D reconstruction trouble during testing. They learn 3D action models with 3D action sequences in training phase and recognize actions with any camera configuration, from single to multiple cameras and from any viewpoint in testing phase. With 3D-key-pose set, this paper also proposes a view-independent framework, which uses 3D data in training only and works with 2D image sequence during testing for the purpose of recognizing actions from arbitrary views. It is different from the existing algorithms that 3D key pose sequence is first obtained using 3D-key-pose set to represent 2D image action sequence, and then, 3D action models are learned with 3D key pose sequence.

Effective representation is very important for action recognition. Johansson [11] reported that with a point-based human model, actions could also be recognized. Human joints contain rich and fine pose information for action recognition. Traditionally, human joints are obtained manually or by a motion capture system that requires markers be attached to the body. Motion capture systems have two major drawbacks: they are obtrusive and expensive [2]. Recently, markerless pose recovery approach is proposed to extract 3D human joints from 3D reconstructed action sequence [12]. Unfortunately, it is very difficult to automatically extract human joints from 2D image action sequence, due to complex poses, self-occlusion, and various viewpoints. In this paper, a new approach is proposed to automatically obtain the 3D human joints of 2D image action sequence using 3D-key-pose set. First, 3D-key-pose set that contains key poses of all actions is selected from 3D human joint sequences of the 3D action samples in training set. Each element of the 3D-key-pose set consists of one 3D point set of subject and its corresponding 3D human joint set. Second, 3D key pose sequence, which matches best with the observation sequence, is selected from the 3D-key-pose set to represent the 2D image action sequence of arbitrary view. 3D key pose sequence contains many discriminative key poses but cannot accurately describe 3D pose of every frame in 2D image action sequence. Considering the above reasons, pose and dynamic of action are modeled respectively in this paper. Exemplar-based embedding and probability of unique key pose are applied to model pose property of 2D image action sequence. Complementary dynamic feature is extracted to model these actions that share the same poses but have different dynamic feature. Finally, these action models are fused to recognize observation sequence from a single or few cameras.

The flow chart of the proposed approach is shown in Fig. 1. This framework includes three parts: selecting 3D-key-pose set from 3D action samples, obtaining 3D key pose sequence using 3D-key-pose set to represent 2D image action sequence, and modeling and recognizing action sequence.

3D-key-pose set. 3D-key-pose set is selected from 3D human joint sequences of 3D training action samples. 3D action samples are obtained from multiple video streams by 3D reconstruction technique. First, multiple video streams are simultaneously captured from multiple static calibrated cameras. Second, foreground/background segmentation is performed on each through background subtraction method. Third, volumetric representation sequence is created with visual hull method [6]. Then, a markerless pose recovery method is adopted to obtain 3D human joint sequence from reconstructed volume data [12]. In 3D space, action sequences can be observed in the same manner as in the real world, and the view difficulty

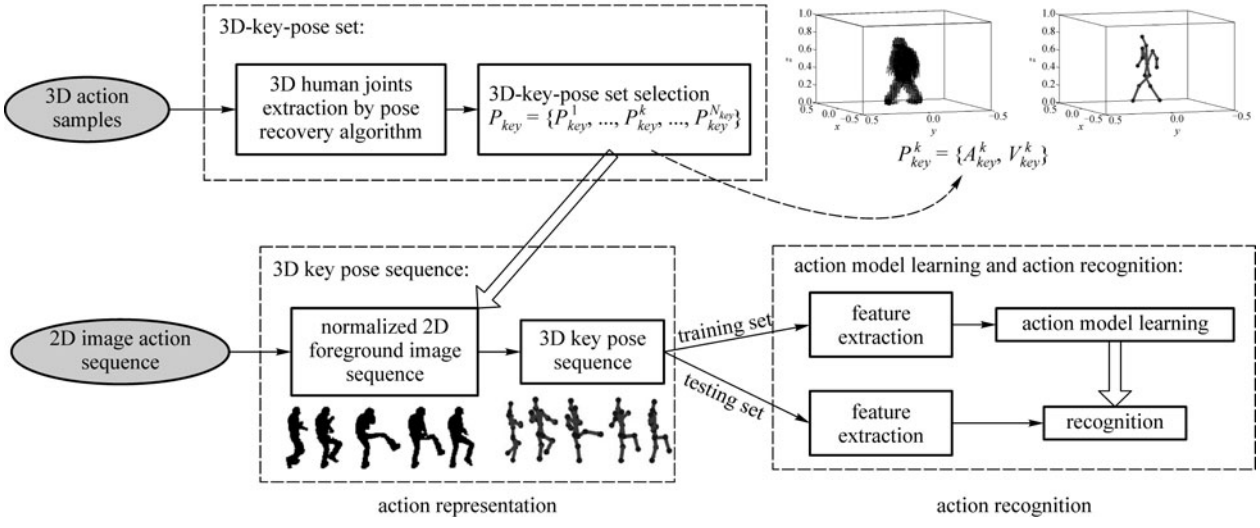


Fig. 1 Flow chart of the proposed action recognition approach using 3D-key-pose set

and self-occluding problem can be easily solved. Finally, elements of 3D-key-pose set are selected one by one from the 3D human joint sequences according to the recognition performance of the training set. In this paper, 3D-key-pose set is used as a bridge linking 2D image action sequence with 3D human joint sequence.

Representation of 2D image action sequence (3D key pose sequence). 3D pose sequence is view independent and an effective action representation method. Unfortunately, it is difficult to extract 3D human pose sequence from 2D image action sequence due to self-occlusion and depth information loss. In fact, humans are able to recognize many actions from several key poses or even single pose, as shown in Figs. 2 and 3. In this paper, the key pose sequence of the original continuous sequence is extracted to represent action sequence. 3D key pose sequence, which matches best with the observation sequence, is selected from the 3D-key-pose set to represent the 2D image action sequence of arbitrary view.

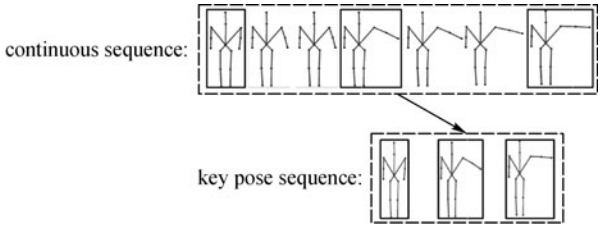


Fig. 2 Key poses in action sequence

Recognition of 2D image action sequence. 3D key pose sequence contains many discriminative key poses but cannot accurately describe the 3D pose of every frame in the 2D image action sequence. Considering the above reasons, exemplar-based embedding method and the probability of unique key pose, which do not model any dynamic information, are applied to model the key pose information of 2D image action sequence. Complementary dynamic feature is extracted from the 3D key pose sequence to model these actions that share the same poses but have different dynamic feature. Finally, these action models are fused to recognize observation sequence from a single or few cameras.

3D volume reconstruction technology and pose recovery method of 3D action are not discussed in detail in this paper. Literature on related field can be found in Refs. [6,12,13]. In addition, the temporal segmentation, which consists of splitting a continuous sequence of motions into elementary meaningful sequences, is accomplished manually. Representation and recognition of 2D image action sequence using 3D-key-pose set are the focuses in this paper.

Section 2 reviews the state-of-the-art methods in view-independent action recognition. Section 3 is action representation that describes the 3D-key-pose set selection approach and 3D key pose sequence extraction using 3D-key-pose set. Section 4 is action recognition that explains the learning algorithm of action models and the design of classifier. Section 5 explains the experimental results. Section 6 concludes this paper.

2 Related works

To allow actions to be learned and recognized using different camera configurations, action descriptions must exhibit some view invariance [8]. Several view-independent action recognition approaches have been brought forward.

The fundamental matrix between two views is a common approach to achieve view invariance in monocular camera configurations. Reference [14] presented an approach for view-invariant human action recognition with inputs of projected 2D human joint positions. Actions were modeled with canonical body poses (consisting of five-ordered joints, two view-invariant values, and a threshold for matching purposes) and trajectories in 2D invariance space called invariance space trajectories. Gritai et al. [15] presented the invariant analysis of human actions on the use of anthropometry to provide constraints on matching. In their study, a point-light display-like representation was used, where a pose was presented through a set of 3D human joints. Nonlinear time warping was used to ensure that similar actions performed at different rates were accurately matched. Yilmaz and Shah [1] proposed a method to perform action recognition in presence of

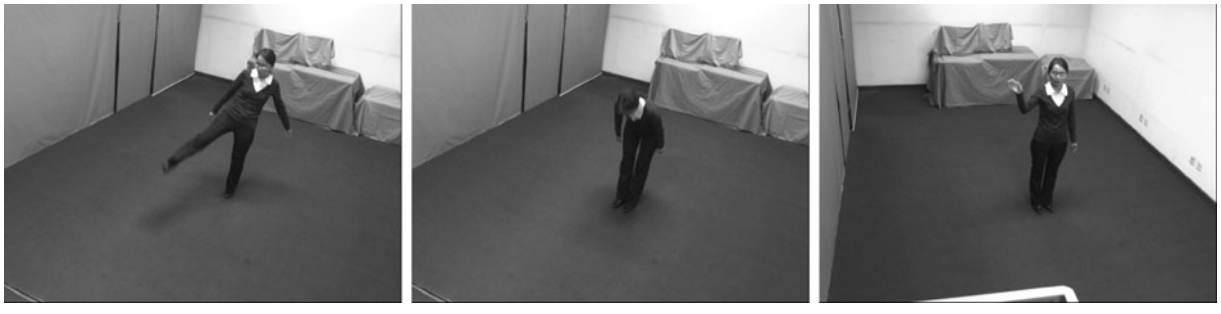


Fig. 3 Single pose in action sequence

camera motion. At each time instant, in addition to the camera motion, a different view of the action was observed. The proposed method was based on the epipolar geometry between any two views. However, instead of relating two static views using the standard fundamental matrix, they modeled the motions of independently moving cameras in the equations governing the epipolar geometry and derived a new relation that was referred to as the "temporal fundamental matrix". In the above approaches, an action was represented as a set of human joints. However, human body joint tracking of monocular action sequence in an unconstrained environment is quiet complex. At present, monocular action recognition methods generally assume human joints tracking has already been performed, and a set of joint trajectories are given.

Multiple cameras configuration is an intuitive solution to deal with the viewpoint changes. It is an important issue to fuse information from multiple cameras. There are two types of fusion methods: learning multiple action models with 2D image action sequences captured in multiple views [4,5,16,17] or learning 3D action model with 3D action sequences [6,12]. For the first methods, a lot of cameras are needed to be set up to capture the action sequences of multiple views. Virtual cameras are often used to reduce hardware cost. For example, Souvenir and Babbs [4] animated the 3D visual hull reconstructed from five cameras and projected the silhouette onto 64 evenly spaced virtual cameras located around the vertical axis of the subject to obtain a larger set of surfaces from various viewpoints for training. For the second methods, about five cameras are enough to simultaneously capture multiple-view action sequences, and then, 3D reconstruction technique is used to obtain 3D action sequence from the multiple action sequences. Weinland et al. [6] used a viewpoint-free motion history volume (MHV) that was extracted from 3D reconstruction volume data to represent action. Fourier-magnitudes and cylindrical coordinates were then used to express the motion templates in an invariant way to the position and rotations around the z-axis. Finally, actions were recognized by the classifier that combines PCA and Mahalanobis distance. Gu et al. [12] represented 3D actions with 3D human joint sequences that were recovered from 3D reconstruction volume data. First, a markerless pose recovery method was adopted to automatically capture the 3D human joint and pose parameter sequences from volume data. Second, multiple configuration features and movement features were extracted from the recovered 3D human joint and pose parameter sequences. Hidden Markov model (HMM) and exemplar-based HMM (EHMM) were then used to model the movement features and configuration features, respectively. Finally, actions were classified by a hierarchical classifier. In 3D space, the viewpoint difficulty and self-occluding problem can be easily solved. The difficult parts of 3D methods are hardware setting and robust 3D reconstruction.

In surveillance and content-analysis scenarios, it is difficult to obtain robust volume data of subjects at present. A new fusion method is proposed to avoid the 3D reconstruction trouble during testing. In training, 3D action models are learned with 3D action sequences. While in testing, actions can be observed with any camera configuration, from single to multiple cameras and from any viewpoint [8]. Weinland et al. [8] presented an action recognition method for arbitrary viewpoint using 3D exemplars. They used EHMM to model actions with 3D volume data. 3D reconstruction was not required during recognition phase; instead, learned 3D exemplars were used to produce 2D image information that was compared to the 2D observation. Yan et al. [9] presented an approach using a four-dimensional (4D) (x, y, z, t) action feature model (4D-AFM) for recognizing actions from arbitrary views. The 4D-AFM encoded shape and motion of subjects observed from multiple views. The modeling process started with reconstructing 3D visual hulls of subjects at each time instant. Spatiotemporal action features were then computed in each view by analyzing the differential geometric properties of spatio-temporal volumes (3D STVs) generated by concatenating the subject's silhouette over the course of the action (x, y, t). These features were mapped to the sequence of 3D visual hulls over time (4D) to build the initial 4D-AFM. Actions were recognized based on the scores of matching action features from the input videos to the model points of 4D-AFMs by exploiting the pairwise interactions of features. Both of the above two methods adopt appearance-based representation method. Human pose-based method is another type of action representation method. 3D human pose recovery is considered as a fundamental step in view-invariant human action recognition. However, inferring 3D poses from a single view is usually very difficult and slow, because parameters to be estimated and recovered are often ambiguous due to the perspective projection [18]. The example-based action recognition system proposed by Lv and Nevatia [18] did not explicitly infer 3D pose at each frame. Instead, from existing action models, they searched for a series of actions that best matched the input sequence. In their approach, each action was modeled as a series of synthetic 2D human poses rendered from a wide range of viewpoints. The existing human poses were rendered from a variety of viewpoints using POSER. The constraints on transition of the synthetic poses were represented by a graph model called Action Net. Given the input, silhouette matching between the input frames and the key poses was performed first using an enhanced Pyramid Match Kernel algorithm. The best matched sequence of actions was then tracked using the Viterbi algorithm.

The approach proposed in this paper also avoids the 3D reconstruction trouble during testing. Different from the above algorithms, in this paper, 3D key pose sequence is first obtained using 3D-key-pose set to represent 2D image

action sequence from arbitrary views. Second, 3D pose feature and dynamic feature extracted from 3D key pose sequence are modeled, respectively. Finally, multiple action models are fused to recognize 2D image action sequences from arbitrary views.

3 Action representation

3D pose sequence is one of the view-invariant action representation methods. However, inferring 3D pose of each frame from arbitrary view action sequence is difficult. Key poses of actions are very discriminative. Humans are able to recognize many actions from several key poses or even one key pose in the action sequence. Using 3D-key-pose set selected from 3D training actions, the 3D key pose sequence of 2D image action sequence are extracted as the view-invariant action representation.

3.1 3D-key-pose set selection

3D-key-pose set is selected from 3D pose sequences of 3D training actions. First, multiple video streams are simultaneously captured from multiple static calibrated cameras. Then, a 3D reconstruction technique is used to obtain 3D action sequence. Finally, a human model-based markerless pose recovery method is adopted to automatically capture the 3D human joint and pose parameter sequences from volume data. When we perform 3D pose recovery from 3D action sequence, the viewpoint difficulty and self-occluding problem can be easily solved. The detailed process of the pose recovery algorithm for 3D actions can be referred to Ref. [12].

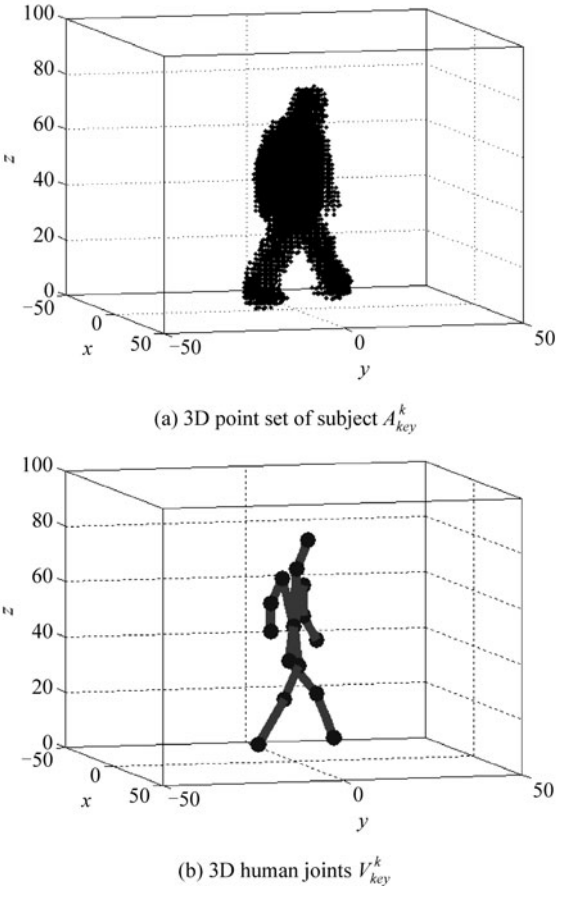
Let $P_{key} = \{P^1_{key},...,P^k_{key},...,P^{N_{key}}_{key}\}$ denote 3D-key-pose set, which includes N_{key} elements. The kth 3D key pose is $P^k_{key} = \{A^k_{key}, V^k_{key}\}$, where $A^k_{key} = \{a_n\}^{N_k}_{n=1}$ is the 3D point set of subject, and $V^k_{key} = \{v_m\}^{15}_{m=1}$ is the corresponding 15 3D human joints, as shown in Fig. 4. The orientation, position, and body height have been normalized.

There are two types of algorithms to select 3D-key-pose set from 3D pose sequence. One algorithm is classifier-independent method, such as clustering method. The other algorithm is classifier-dependent method, which selects 3D key poses from 3D action samples based on the recognition performance of classifier. The second algorithm is adopted in this paper. 3D key poses are selected one by one from 3D action sequences according to the recognition rate of training set. Let $\zeta = \{y_i\}^{M_\zeta}_{i=1}$ be the set of all frames in training action sequences, where y_i includes the 3D point set of subject and corresponding 3D human joints. The feature of 3D action sequence for classification is 3D human joint sequence. Exemplar-based embedding algorithm [19] is used to model and recognize 3D action sequence. Details of the exemplar-based embedding algorithm are explained in the next section of this paper. Table 1 shows the steps of selecting 3D-key-pose set.

In one experiment, the first 24 key poses selected with this algorithm are shown in Fig. 5. These 3D key poses do not only contain discriminative poses but also common poses shared by several classes.

3.2 3D key pose sequence extraction

2D image action sequence is captured from static camera. Foreground/background segmentation is then performed



(a) 3D point set of subject A^k_{key}

(b) 3D human joints V^k_{key}

Fig. 4 Example of 3D key pose

Table 1 3D-key-pose set selection algorithm

Select 3D-key-pose set: $P_{key} = \left\{P^1_{key},...,P^k_{key},...,P^{N_{key}}_{key}\right\}$	
Step 1	Set 3D-key-pose set $P_{key} = \varnothing$;
Step 2	Find $y^* \in \zeta$ and $y^* \notin P_{key}$, where exemplar-based embedding classifier using exemplar set $\{P_{key} \cup \{y^*\}\}$ has the best recognition performance in training set; if multiple y^* are with same performance, we will randomly pick one. $P_{key} = \{P_{key} \cup \{y^*\}\}$;
Step 3	Repeat Step 2 until N_{key} key poses are selected or the recognition rate of training set converges to a stable value.

through background subtraction method. 3D key poses, which are the most similar to silhouette image sequence, are chosen from 3D-key-pose set to be the 3D key pose sequence of 2D image action sequence. The sketch map of 3D key pose sequence extraction from 3D-key-pose set is shown in Fig. 6.

Let $Y=\{y_1,...,y_t...,y_T\}$ be the binary silhouette sequence of 2D image action sequence. Let $P_{key}(Y)=\{P_{key}^{k_1},...,P_{key}^{k_t},...,P_{key}^{k_T}\}$ denote the 3D key pose sequence that is chosen from 3D-key-pose set, where k_t is obtained as follows:

$$k_t=\arg\min_{k=1}^{N_{key}} D(y_t,P_{key}^k), \tag{1}$$

where $D(y_t,P_{key}^k)$ is the distance between 2D silhouette image y_t and 3D key pose P_{key}^k. To calculate this distance, 3D key poses are projected into several viewpoints, as shown in Fig. 7. The projection plane of camera is assumed to be approximately perpendicular to the ground. Therefore, projection viewpoints only change around the vertical axis. Let $p_m^k(1\leqslant m\leqslant M)$ be the mth binary projection image of 3D key pose P_{key}^k. Then, the distance between silhouette image y_t and 3D key pose P_{key}^k can be defined as follows:

$$D(y_t,P_{key}^k)=\min_{m=1}^{M} d(y_t,p_m^k). \tag{2}$$

The distance measurement between silhouette image y_t and 3D key pose P_{key}^k should be subject-free and viewpoint-free. In the following, some factors, such as normalization of image, selection of distance definition, confidence weight, and fusion of multiple viewpoints, are discussed in detail.

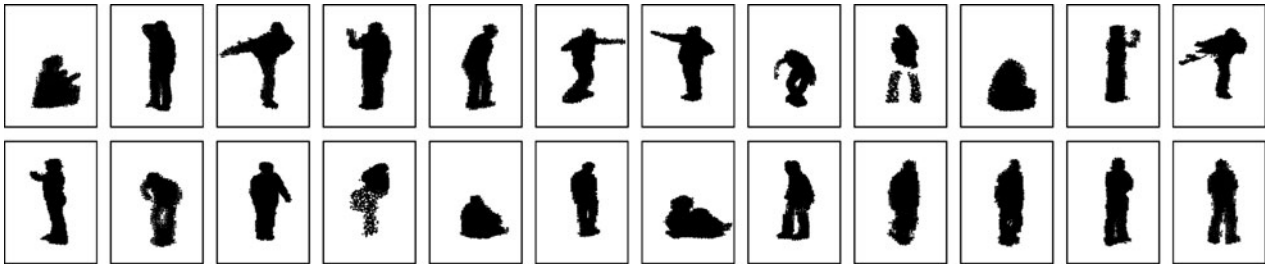


Fig. 5 First 24 3D key poses as returned by the algorithm in Table 1 (only the 3D point sets of subject are shown)

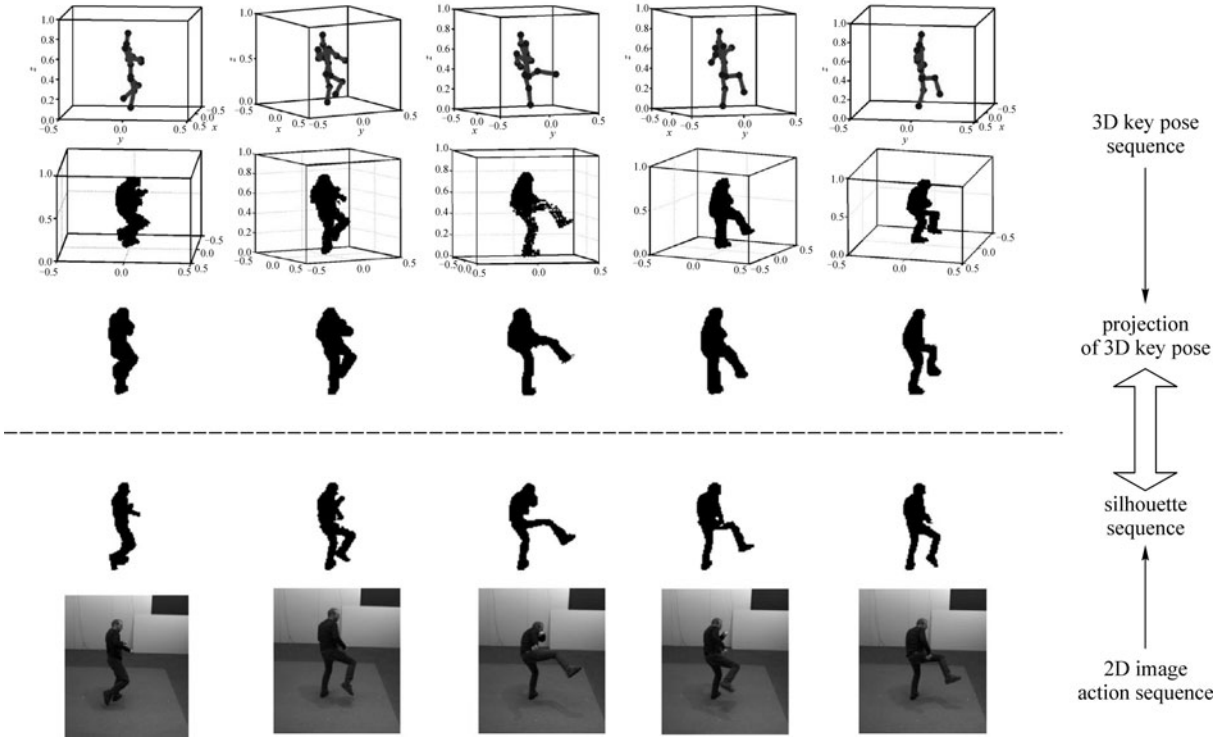


Fig. 6 Sketch map of 3D key pose sequence extraction from 3D-key-pose set

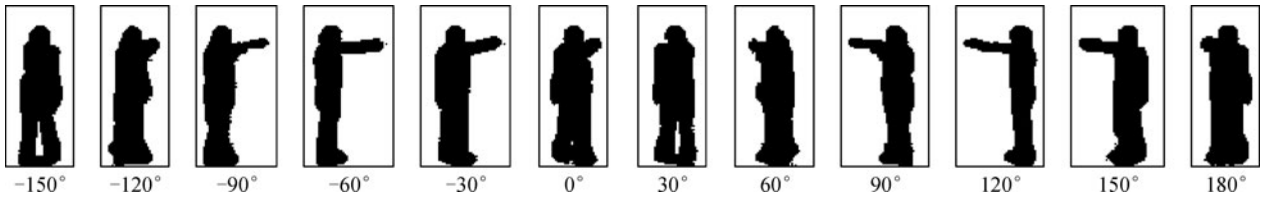


Fig. 7 Projection images of one 3D key pose

3.2.1 Normalization of silhouette image

Size normalization of silhouette image is used to adapt to the change of subjects' sizes. Let $S = \left\{\left(s_x^{(n)}, s_y^{(n)}\right)\right\}_{n=1}^{N_S}$ be the set of silhouette points, where $\left(s_x^{(n)}, s_y^{(n)}\right)$ are the image coordinates of the nth silhouette point. The body height of subject is normalized to 60 pixels, and the silhouette image is normalized to 80×60 pixels. The process of normalization method is shown in Fig. 8. The detailed procedure is listed in Table 2, where the zoom parameter, $z = 60 \Big/ \left(\max_{n=1}^{N_S}\left\{s_y^{(n)}\right\} - \min_{n=1}^{N_S}\left\{s_y^{(n)}\right\}\right)$, is obtained with the first frame of silhouette sequence. The subject is made to assume a "standing upright" pose in the first frame.

Normalization according to the subject's height cannot adapt to the variance of body type (fat or thin). Therefore, we should select distance measurements that are insensitive to body type and sensitive to pose change.

3.2.2 Distance measurement

There are two types of distance measurements between two binary images I_R and I_T. One compares the difference of image, such as the squared Euclidean distance, $d_E(I_T, I_R) = \frac{1}{|N_{I_R}|}\sum_x \sum_y |I_R(x,y) - I_T(x,y)|$, which is computed between the vector representations of the binary silhouette images. Hence, this distance is simply the number of pixels with different values in both images. The other calculates the distance of shape denoted by silhouette point set, such as Chamfer distance. Let R and T be two silhouette point sets of two binary images. Then, the Chamfer distance between R and T is defined as

$$d_C(T,R) = \frac{1}{|T|}\sum_{t \in T} dd(t,R) = \frac{1}{|T|}\sum_{t \in T} \min_{r \in R}\|t - r\|. \quad (3)$$

Moreover, the undirected Chamfer distance is defined as

$$\overline{d}_C(T,R) = \frac{d_C(T,R) + d_C(R,T)}{2}. \quad (4)$$

Compared with squared Euclidean distance, the undirected Chamfer distance does not only consider the different points between test point set and reference point set but also consider the distance relationship of these different points. The distance we consider is the undirected Chamfer distance of silhouette images, as in formula (4).

Undirected Chamfer distance of silhouette images is sensitive to pose change and can tolerate some body type change. Figure 9 shows the undirected Chamfer distances between normal "stand" pose (body type parameter is 1.0) and other "stand" poses (body pose parameter is from 0.5 to 2.0), and the undirected Chamfer distances between normal "stand" pose and other poses. Most of the distances between the same "stand" poses in different body types are smaller than those between different poses. The undirected Chamfer distance, which is insensitive to body type change and sensitive to pose change, helps to adapt different subjects.

3.2.3 Confidence weight

Due to self-occlusion, projection images of 3D key pose at different viewpoints have different ability to discriminate this 3D pose. At some viewpoints, the movement of body parts can be observed completely, while at other viewpoints, the movement of body parts cannot be observed. As shown in Fig. 7, the lifting arm can be perfectly observed

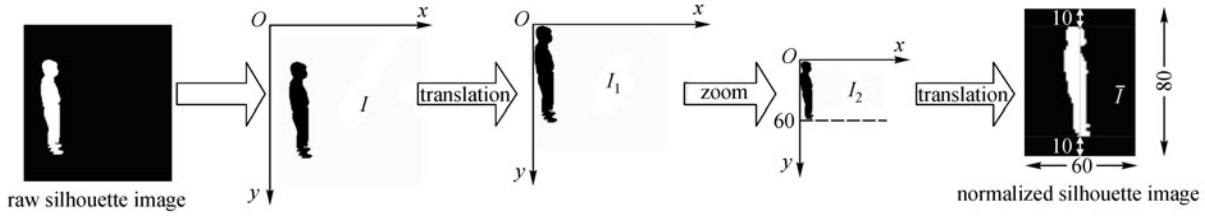


Fig. 8 Flow chart of normalization of silhouette image

Table 2 Normalization method of silhouette image

Step 1	Translate point set $S=\left\{\left(s_x^{(n)},s_y^{(n)}\right)\right\}_{n=1}^{N_S}$ to the origin point to obtain $S_1=\left\{\left(s_{x_1}^{(n)},s_{y_1}^{(n)}\right)\right\}_{n=1}^{N_S}$, where $s_{x_1}^{(n)}=s_x^{(n)}-\min_{n=1}^{N_S}\left\{s_x^{(n)}\right\}$, $s_{y_1}^{(n)}=s_y^{(n)}-\min_{n=1}^{N_S}\left\{s_y^{(n)}\right\}$;
Step 2	Zoom height of subject to 60 pixels to obtain $S_2=\left\{\left(s_{x_2}^{(n)},s_{y_2}^{(n)}\right)\right\}_{n=1}^{N_S}$, where $s_{x_2}^{(n)}=z\times s_{x_1}^{(n)}$, $s_{y_2}^{(n)}=z\times s_{y_1}^{(n)}$, and z is the zoom parameter;
Step 3	Normalize the silhouette image to 80 × 60 pixels. Let $\overline{S}=\left\{\left(\overline{s}_{x_1}^{(n)},\overline{s}_{y_1}^{(n)}\right)\right\}_{n=1}^{N_S}$ be the normalized silhouette image. Then, $\overline{s}_x^{(n)}=s_{x_2}^{(n)}+30$, $\overline{s}_y^{(n)}=s_{y_2}^{(n)}-\max_{n=1}^{N_S}\left\{s_{y_2}^{(n)}\right\}+70$.

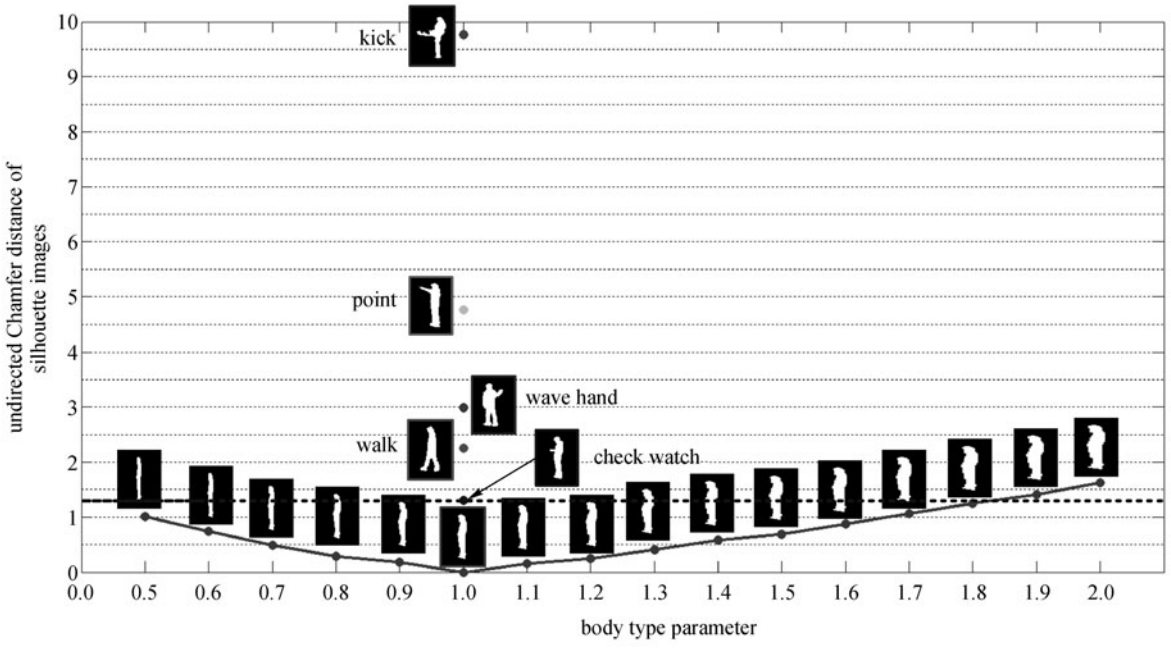


Fig. 9 Undirected Chamfer distance of silhouette images and body type parameter

in some projection images, but it cannot or partly be observed in other projection images due to self-occlusion. To reduce the affect of self-occlusion, confidence weight is adopted in this paper. For each projection image of every 3D key pose, a confidence weight is calculated to present the ability to discriminate this 3D key pose from other poses.

Let ω_m^k denote the confidence weight of the projection image $p_m^k(1\leqslant m\leqslant M)$, where p_m^k is the mth binary projection image of 3D key pose P_{key}^k. At these viewpoints where the movement of body parts can be observed, the projection image is more different from the other poses. According to the distances between projection images, the confidence weights are calculated. The detailed process is listed in Table 3.

At these viewpoints, where the movement of body parts can be observed, the confidence weights are bigger. At the other viewpoints where the movement of body parts is self-occluded, the confidence weights are smaller. Figure 10 shows the confidence weights of two 3D key poses, "stand" and "point". The projection images of "stand" at different viewpoints are similar, so the confidence weights of "stand" are little different. For the 3D key pose "point", the confidence weights at these viewpoints where the movement of body parts can be observed are bigger than those at other viewpoints.

Then, the distance between silhouette image y_t and 3D key pose, P_{key}^k, can be redefined as follows:

$$D_\omega(y_t,P_{key}^k)=\min_{m=1}^{M}\left\{\frac{1}{\omega_m^k}d(y_t,p_m^k)\right\}. \tag{5}$$

3.2.4 Fusion of multiple cameras

The observations from multiple cameras can easily be incorporated. Assuming multiple view observations, $Y=\{Y^r\}_{1\leqslant r\leqslant R}$, where $R(R\geqslant 1)$ is the count of cameras,

and $Y^r = \{y_1^r,...,y_t^r,...,y_T^r\}$ denotes the rth binary silhouette sequence, we can write the 3D key pose sequence of this observations as $P_{key}(Y) = \{P_{key}^{k_1},...,P_{key}^{k_t},...,P_{key}^{k_T}\}$, where k_t is obtained as follows:

$$k_t = \arg\min_{k=1}^{N_{key}} \sum_{r=1}^{R} D_\omega\left(y_t^r, P_{key}^k\right). \quad (6)$$

3D key pose sequence is view-independent and effective action representation method. In the next section, fusion of multiple action models is proposed to recognize action sequence represented by 3D key pose sequence.

4 Action recognition

3D key pose sequence contains many discriminative key poses but does not accurately describe 3D pose of every frame in 2D image action sequence. Considering the above reasons, pose and dynamic properties of action sequence are modeled, respectively, as shown in Fig. 11. Exemplar-based embedding and probability of unique key pose, which do not model dynamic information, are applied to model the pose information of 2D image action sequence. Moreover, complementary dynamic feature is extracted to model these actions that share the same poses but have different dynamic feature, such as "sit down" and "get up".

4.1 Exemplar-based embedding

Weinland et al. [19] introduced exemplar-based embedding method to model single fixed-view action sequence. Their method had recognition rates that equaled or exceeded those of state-of-the-art approaches. Exemplar-

Table 3 Confidence weight calculation

Step 1	Calculate the undirected Chamfer distances between projection image, p_m^k, and projection images of other 3D key poses: $D(p_m^k,p_n^i) = \overline{d}_C(p_m^k,p_n^i)$ $(1 \leqslant k,i \leqslant N_{key},\ 1 \leqslant m,n \leqslant M,\ i \neq k)$;
Step 2	Obtain weight-related matrix: $DD(k,m) = \min_{\substack{1 \leqslant i \leqslant N_{key}, i \neq k \\ 1 \leqslant n \leqslant M}} D\left(p_m^k,p_n^i\right)$, where $DD(k,m)$ is the smallest distance between p_m^k and projection images of other 3D key poses;
Step 3	Normalize weight-related matrix: $\overline{DD}(k,m) = DD(k,m) / \max_{\substack{1 \leqslant i \leqslant N_{key} \\ 1 \leqslant n \leqslant M}} DD(i,n)$;
Step 4	Normalize the weight-related values of the same 3D key pose: $\overline{DD^{(k)}}(k,m) = \overline{DD}(k,m) - \max_{1 \leqslant m \leqslant M} \overline{DD}(k,m) + 1\ (1 \leqslant k \leqslant N_{key})$;
Step 5	Obtain confidence weight: $\omega_m^k = 1 + \exp\left[\alpha \times \overline{DD^{(k)}}(k,m)\right]$ $(\alpha > 0)$.

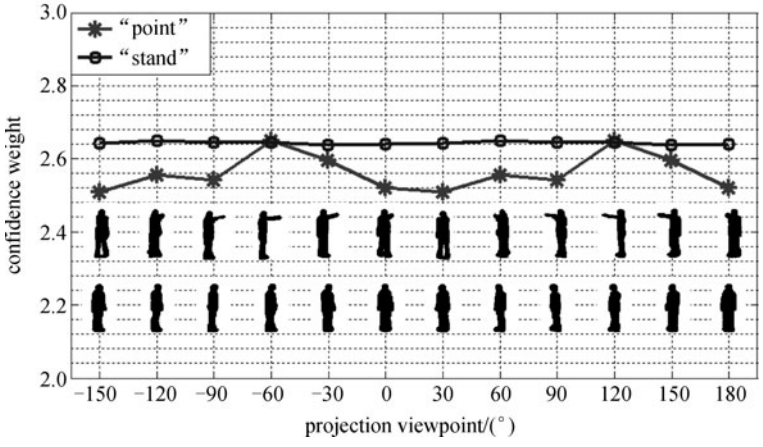


Fig. 10 Examples of confidence weights

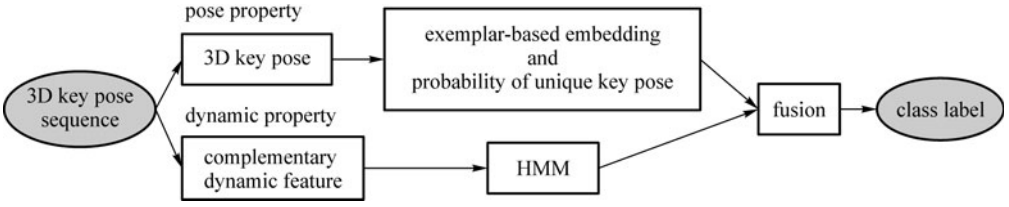


Fig. 11 Flow chart of action recognition

based embedding method is based on simple matching of exemplars to image sequences and does not account for dynamics. This paper use exemplar-based embedding to model the pose property of 2D image action sequence represented with 3D key pose sequence.

3D key pose sequence $P_{key}(Y) = \{P_{key}^{k_1},...,P_{key}^{k_t},...,P_{key}^{k_T}\}$ is selected from 3D-key-pose set $P_{key} = \{P_{key}^{1},...,P_{key}^{k},...,P_{key}^{N_{key}}\}$ to represent 2D image action sequence $Y = \{y_1,...y_t,...,y_T\}$. Then, every element of 3D key pose sequence $P_{key}^{k_t}$ contains two 3D datasets: 3D human joint set, $V_{key}^{k_t} = \{v_m\}_{m=1}^{15}$, and 3D point set of subject, $A_{key}^{k_t} = \{a_n\}_{n=1}^{N_{k_t}}$. Let $Y_{V_{key}} = \{V_{key}^{k_1},...,V_{key}^{k_t},...,V_{key}^{k_T}\}$ denote the 3D human joint sequence of 3D key pose sequence. Let $X = \{x^1,...,x^k,...,x^{N_X}\}$ be the exemplar set. Because all samples of 3D key pose sequences are selected from the 3D-key-pose set, the elements of exemplar set are directly chosen from the 3D-key-pose set, that is, $x^k = V_{key}^{k}$. The exemplar-based embedding starts by computing for each exemplar x^k the minimum distance to frames in the sequence:

$$d_k^*(Y) = \min_{t=1}^{T} d(x^k, V_{key}^{k_t}), \tag{7}$$

where d is the Euclidean distance between the primitives considered. Then, we obtain the feature vectors that result from concatenating all the minimum distances:

$$D^*(Y) = [d_1^*(Y),...,d_{N_X}^*(Y)]^{\mathrm{T}} \in \Re^{N_X}. \tag{8}$$

The process of feature vector extraction is shown in Fig. 12.

After feature vectors are extracted from 3D human joint sequences, each action class is modeled through a single Gaussian distribution: $p(D^*|c) = \aleph(D^*|\mu_c,\Sigma_c)$, where c is the action label, and the model parameter $\lambda_c = \{\mu_c,\Sigma_c\}$ is learned through maximum a posteriori estimations with the training samples.

4.2 Probability of unique key poses

Some key poses in 3D-key-pose set can be used to recognize action class only with one of them. We call these key poses as unique key poses, as shown in Fig. 13. Another key poses in 3D-key-pose set are shared by more than one action class and cannot classify actions. These key poses are called common key poses, as shown in Fig. 14.

Let D_{key}^c denote the unique key poses that belong to the cth action class. $D_{key}^c(Y)$ is the occupancy frame number of D_{key}^c in the 3D key pose sequence $P_{key}(Y) = \{P_{key}^{(k_1)},..., P_{key}^{(k_t)},...,P_{key}^{(k_T)}\}$. $p\left(P_{key}(Y)/c\right)$ is defined as the occupancy probability of unique key poses of the cth action class. It is defined as

$$p\left(P_{key}(Y)/c\right) = D_{key}^c(Y)/T, \tag{9}$$

where T is the length of action sequence. The occupancy probability of unique key poses is closely related to action recognition. The occupancy probability of unique key poses makes more use of pose property and helps to improve the recognition performance.

4.3 Complementary dynamic feature

Exemplar-based embedding and probability of unique key poses only model the pose property of action sequence. However, it is the fact that not all actions can be discriminated without dynamics. A typical example is an action and its reversal, such as “sit down” and “get up”. Without taking temporal ordering into account, it will be very difficult to discriminate them. To recognize such actions, a modeling of complementary dynamic feature is required coupled with these pose features. In the experimental datasets of this paper, some actions, “sit down”, “get up”, and “pick up”, cannot be well discriminated by pose features but can be discriminated by body height

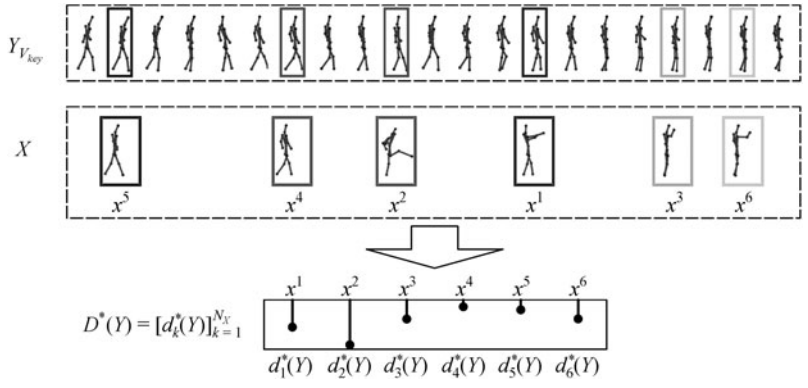


Fig. 12 Feature extraction sketch of exemplar-based embedding

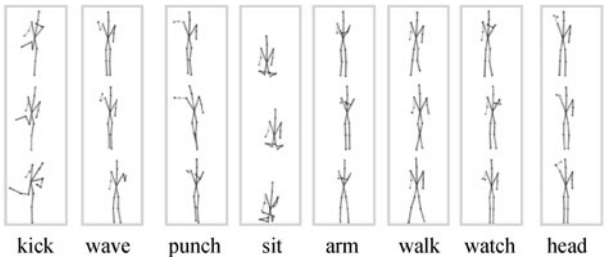


Fig. 13 Samples of unique key poses

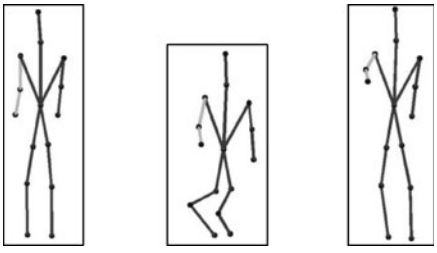


Fig. 14 Samples of common key poses

sequence. Therefore, body height sequence, H, is extracted as the complementary dynamic feature and modeled through hidden Markov model (HMM) [20] denoted by $\lambda_c^H = \{A,b,\pi\}$. For other datasets, complementary dynamic feature should be selected according to the characteristics of action and its reversal.

4.4 Classification fused multiple models

Two pose models and one dynamic model are learned. $\lambda_c = \{\mu_c,\Sigma_c\}$ is the Gaussian model parameter of exemplar-based embedding, $p\left(P_{key}(Y)/c\right)$ is the occupancy probability of unique key poses, and $\lambda_c^H = \{A,b,\pi\}$ is the parameter of HMM of body height sequence. These action models are fused to recognize actions.

Testing action sequence is classified with a two-layer classifier that fuses multiple models. The first layer is a MAP classifier that fuses two pose models as

$$\begin{aligned} c_1^* &= \arg\max_c \left\{p\left(c|D^*(Y),P_{key}(Y)\right)\right\} \\ &= \arg\max_c \left\{p\left(D^*(Y)/\lambda_c\right) \times p\left(P_{key}(Y)/c\right) \times p(c)\right\}. \end{aligned} \tag{10}$$

If the decision of the first layer belongs to body height-changing actions, action sequence will be recognized by the second MAP classifier with body height features as

$$\begin{aligned} c_2^* &= \arg\max_c \left\{p\left(c|D^*(Y),P_{key}(Y),H(Y)\right)\right\} \\ &= \arg\max_c \left\{p\left(D^*(Y)/\lambda_c\right) \times p\left(P_{key}(Y)/c\right) \right. \\ &\quad \left. \times p\left(H(Y)/\lambda_c^H\right) \times p(c)\right\}. \end{aligned} \tag{11}$$

5 Experimental results and analysis

5.1 Dataset

This paper presents results on motion dataset from IXMAS dataset[1]. The dataset provides raw videos, silhouettes, and 3D volume sequences. These actions include "check watch", "cross arm", "scratch head", "sit down", "get up", "turn around", "walk in a circle", "wave hand", "punch", "kick", and "pick up", as shown in Fig. 15. There are 12 subjects with different body type, as shown in Fig. 16. Each subject plays each action three times. Image resolution is 390 × 291 pixels, and the volume of interest is divided into 64 × 64 × 64 voxels. To demonstrate view invariance, subjects freely change their orientations.

The acquisition is achieved using five standard FireWire cameras. One of the cameras is under overlooking viewpoint. The projection plane of this camera is not perpendicular to the ground. Therefore, in the following experiments, we do not consider this camera. The images simultaneously captured from other four cameras are shown in Fig. 17.

Leave-one-out cross-validation experimental method is used. Eleven of the 12 subjects in the dataset are used for action model learning and 3D-key-pose set selection, and the 12th subject is used for evaluation. This procedure is repeated 12 times by permuting test subject.

5.2 3D-key-pose set selection

According to the recognition performance, elements of 3D-key-pose set are selected from the 3D action samples one by one. Exemplar-based embedding is used to model and recognize 3D action sequences. Figure 18 shows the average recognition rates versus count of 3D key poses. The recognition rates of training set and testing set tend to stable values with the increasing of key pose. The count of 3D key poses is set 480 in our experiments. Then, the recognition rates of training set and testing set are 100% and 93%, respectively.

5.3 3D key pose sequence

Using 3D-key-pose set, 3D key pose sequence $P_{key}(Y) = \{P_{key}^{k_1},\ldots,P_{key}^{k_t},\ldots,P_{key}^{k_T}\}$ is obtained to represent 2D image

1) The multiple-video data used here are from INRIA Rhône-Alpes' multiple-camera platform Grimage and PERCEPTION research group. The database is available at https://charibdis.inrialpes.fr.

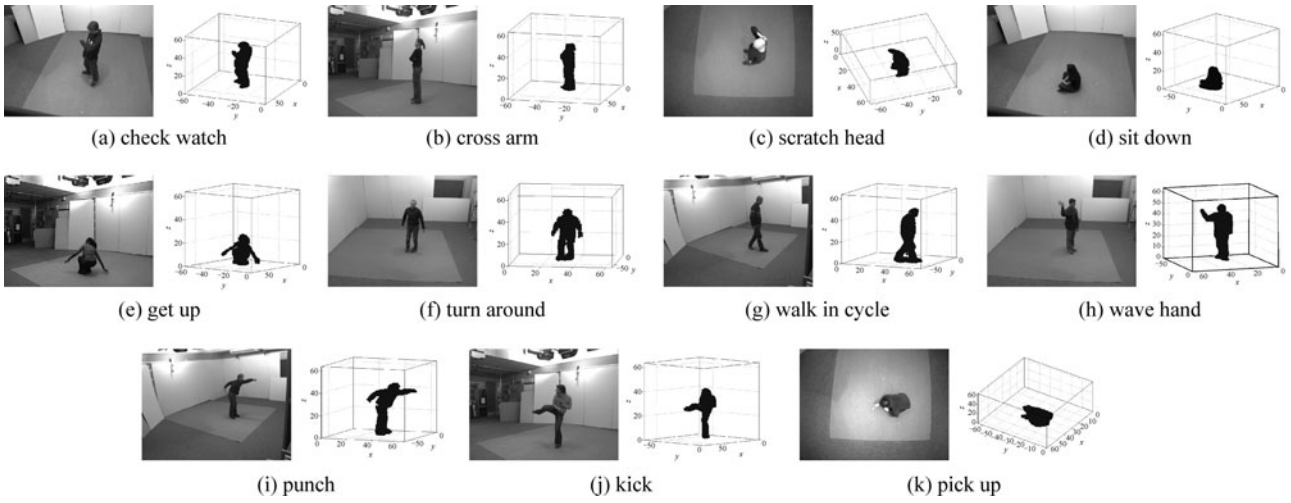


Fig. 15 Samples of images and 3D volume data in IXMAS dataset

Fig. 16 Twelve subjects in IXMAS dataset

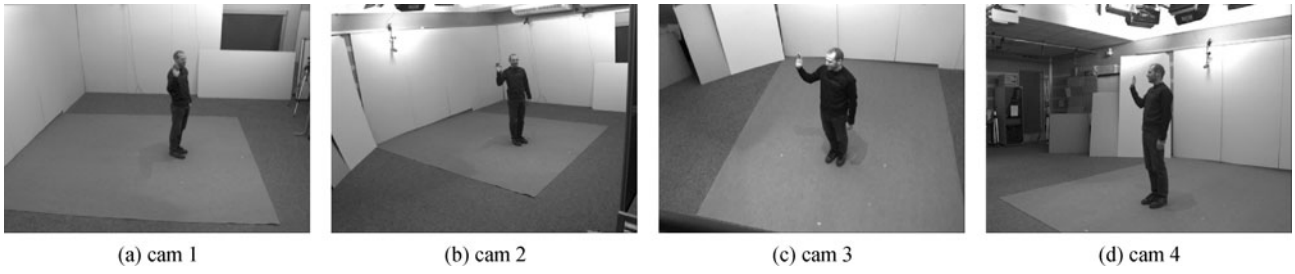


Fig. 17 Images simultaneously captured from four cameras

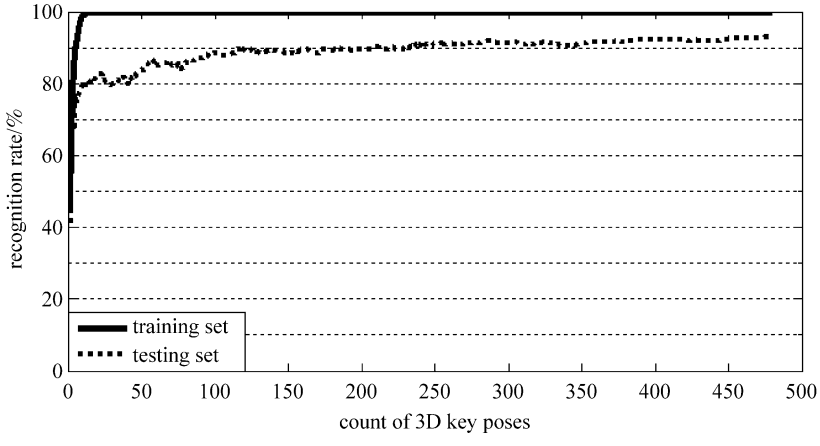


Fig. 18 Recognition rate versus count of 3D key poses

action sequence *Y*.

5.3.1 3D key pose sequence with single camera

Extracting 3D poses from single-view 2D image sequence is very important for many applications, where there is only one camera.

Figure 19 shows examples of 3D key poses extracted from single-view action sequences in one experiment. One frame of every action class is shown. In each sub-image, the top line shows the result of cam 3, and the bottom line shows the result of cam 4. Moreover, each line shows the silhouette of 2D action, 3D point set of the selected 3D key pose, and 3D human joints of the selected 3D key pose, respectively. The 3D key poses of the same frame obtained from cam 3 and cam 4 are always different. Pitch angle of cam 3 is larger than cam 4. However, our algorithm applies to where the projection plane of camera is perpendicular to the ground. Then, there are many mistakes of the 3D key poses of cam 3. The results of cam 4 are better than those of cam 3.

Due to self-occlusion or large pitch angle, 3D key pose of single camera may be selected wrongly, as shown in Fig. 20(a). The 3D key pose of the same frame obtained by four cameras fusion is shown in Fig. 20(b). Through fusion of multiple cameras, the algorithm performance can be improved. The fusion results of multiple cameras are explained in detail in the next subsection.

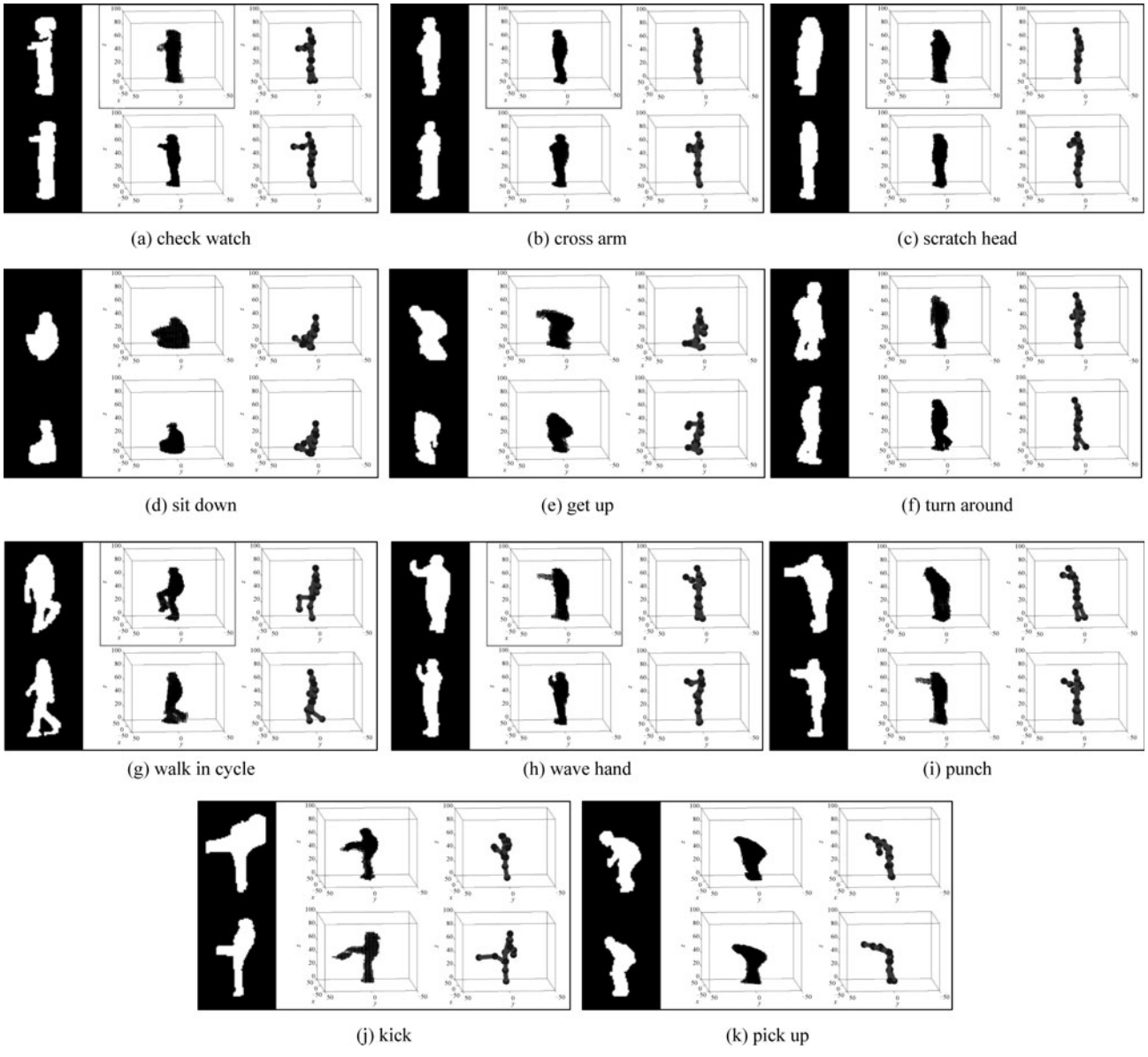


Fig. 19 Examples of 3D key pose sequences (single camera)

5.3.2 3D key pose sequence with multiple cameras

Fusion of multiple cameras can reduce the effect of self-occlusion and adapt to the change of orientation of subject. The 3D key pose sequences obtained from multiple cameras are shown in Fig. 21. One frame of every action class is shown. In each sub-image, the top line shows the silhouettes of 2D action from four cameras, and the bottom line shows a 3D point set of the selected 3D key pose and 3D human joints of the selected 3D key pose, respectively. Although the movement body parts are not observed in some cameras due to self-occlusion (such as the third and the fourth camera in Fig. 21(c)), the correct 3D key pose can be selected through fusion of multiple cameras. In addition, the fusion of multiple cameras also reduces the affect of noise (such as Fig. 21(b)).

5.3.3 Effect of confidence weight

Figure 22 compares the results of the same "stand" pose with confidence weight and without confidence weight. Because the projection images of 3D "wave" pose in some

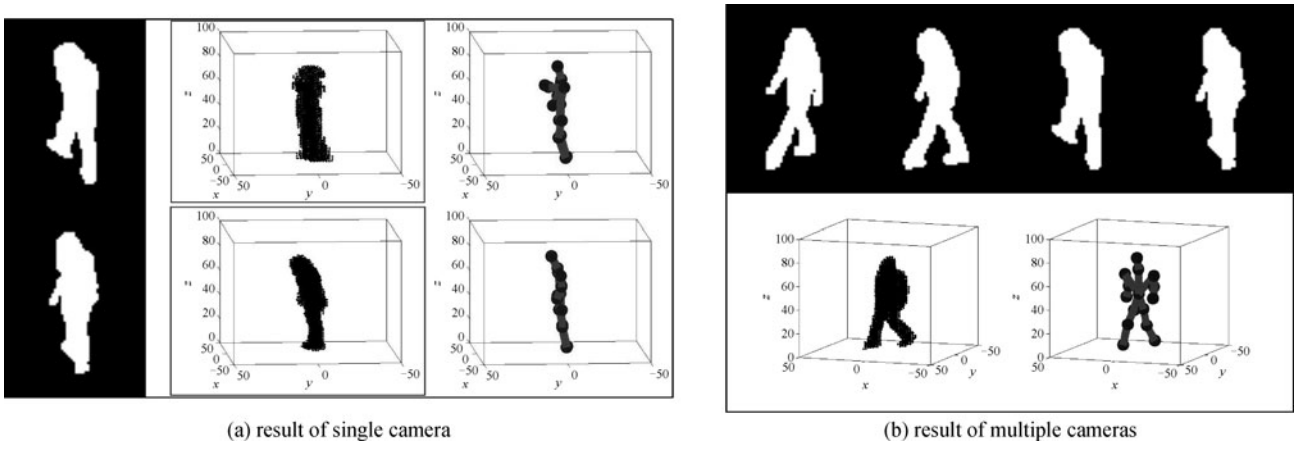


Fig. 20 Examples of 3D key pose (single camera and multiple cameras)

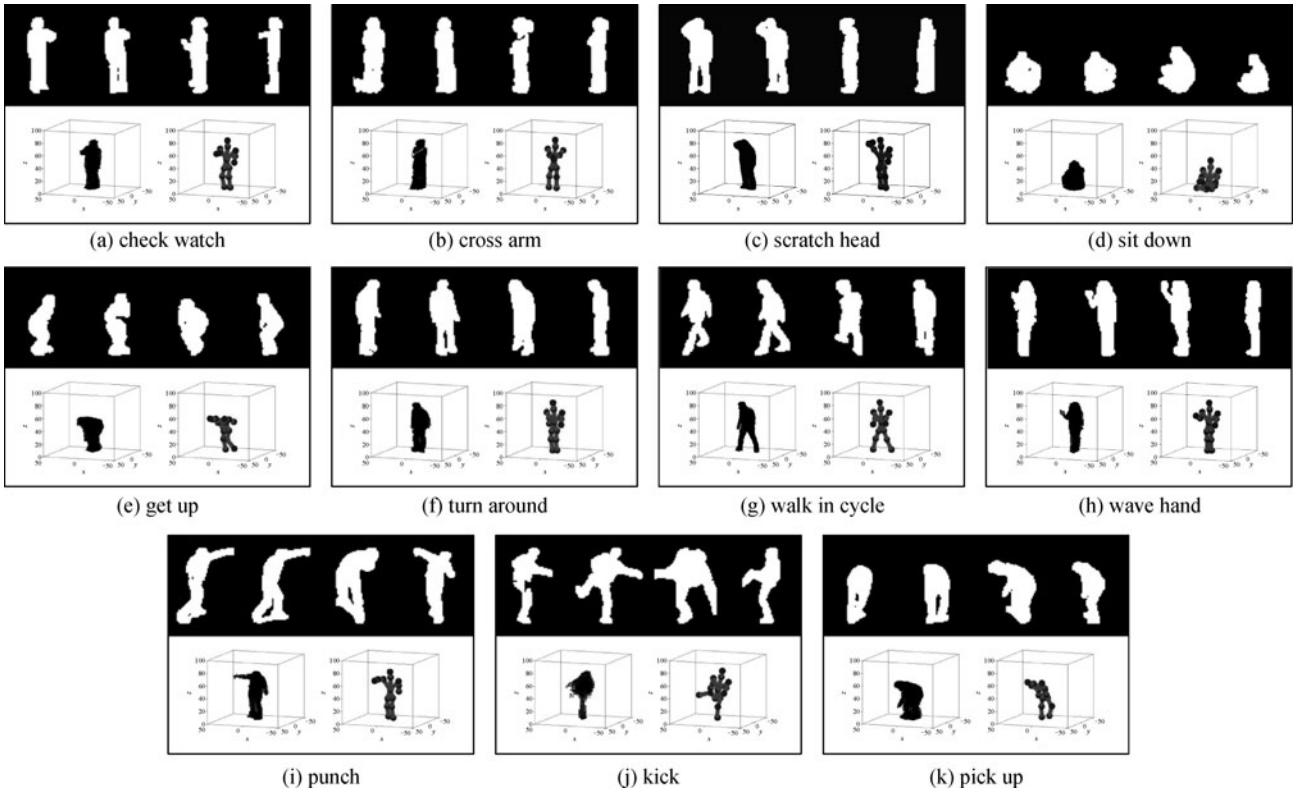


Fig. 21 Examples of 3D key pose sequences (four-camera fusion)

viewpoints are similar to the "stand" silhouettes, the 3D key pose of "stand" pose selected is "wave" pose without confidence weight. With confidence weight, the correct 3D "stand" pose is selected. Confidence weight reduces the confusion of projection images of 3D key poses due to self-occlusion and improves the algorithm performance.

5.4 Recognition rates

The recognition rates per camera are given in Fig. 23. The average recognition rates of cam 1, cam 2, cam 3, and cam 4, are 78%, 78%, 67%, and 78%, respectively. The cam 3 scores worst. The pitch angle of cam 3 is bigger than those of other cameras. The pitch angle has an impact on the recognition performance.

For 2D image action recognition captured from single camera with arbitrary subject orientations, the movement of body parts may be self-occlusion. Self-occlusion has an important impact on the recognition performance, and results in that the recognition rates of single camera are not as good as we expect. In the next experiment, several cameras are used in conjunction to test camera combinations. The recognition rates of camera combinations are listed in Table 4. More cameras are used, more high recognition rates are. When all of the four cameras are fused, the recognition rate reaches 93%.

Comparison between the approach of this study and previous approaches are given in Table 5. The first three algorithms listed in this table deal with actions in fixed single viewpoint using one camera, and they reach very high recognition rates. For the other approaches listed in this table, subjects can change their orientations freely. Some approaches whose camera counts are "3D" are 3D action recognition methods. The recognition rates of 3D action are a little lower than those with fixed subject orientation. The other approaches in this table recognize

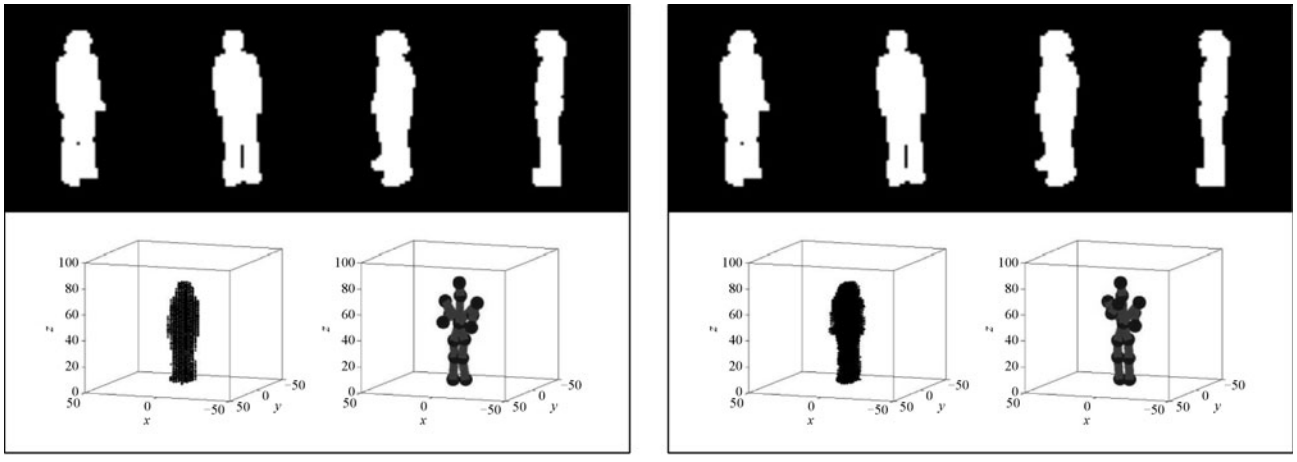


Fig. 22 Effect of confidence weight (four-camera fusion)

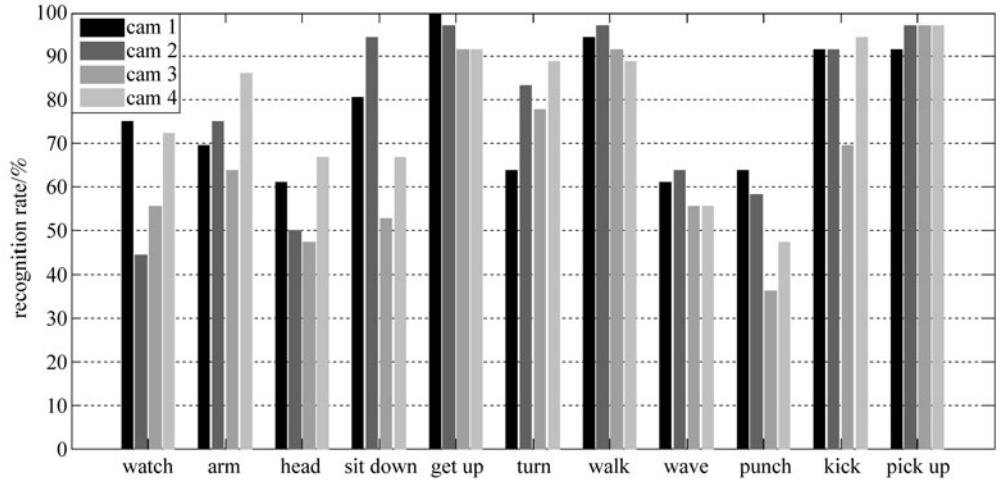


Fig. 23 Recognition rates per camera

Table 4 Recognition rates versus camera count

camera count	camera label	recognition rate
1	cam 1	78%
	cam 2	78%
	cam 3	67%
	cam 4	78%
	average recognition rate	**75%**
2	cam 1, cam 2	84%
	cam 1, cam 3	82%
	cam 1, cam 4	87%
	cam 2, cam 3	80%
	cam 2, cam 4	88%
	cam 3, cam 4	80%
	average recognition rate	**83%**
3	cam 1, cam 2, cam 3	87%
	cam 1, cam 2, cam 4	88%
	cam 1, cam 3, cam 4	88%
	cam 2, cam 3, cam 4	85%
	average recognition rate	**87%**
4	cam 1, cam 2, cam 3, cam 4	93%

arbitrary view actions. The recognition rates of these approaches are lower than those with fixed subject orientation or 3D actions. It is difficult to compare all of these approaches since database and environments are different. However, the results can give a general overview and comparison of approaches in action recognition. Compared with the mentioned research, the approach proposed yields better results. When observations captured from multiple cameras are fused, the recognition performance of proposed algorithm is comparable with those of 3D action recognition methods.

Three algorithms in Table 5 learn 3D models for recognizing action sequence from a single or few cameras and use the same dataset (IXMAS dataset) as our approach. We list the detailed comparison of these algorithms in Table 6. Weinland et al. [8] modeled actions using 3D occupancy grids in an exemplar-based HMM to recognize those same actions from a single or few cameras. Learned 3D exemplars were used to produce 2D image information that was compared to the observations (2D silhouette sequence). Yan et al. [9] used 4D (x, y, z, t) action feature model (4D-AFM) that encoded shape and motion of actors for recognizing actions from arbitrary views. Actions were recognized based on the scores of matching spatiotemporal action features from the input videos to the model points of 4D-AFMs by exploiting the pair-wise interactions of features. Liu et al. [10] used two types of features: a quantized vocabulary of local spatio-temporal (ST) volumes (or cuboids) and a quantized vocabulary of spin-images, which aims to capture the shape deformation

Table 5 Comparison of the proposed approach with previous researches

algorithms	action count	subject count	database	representation	camera count	recognition rates
Wang et al. [21]	10	9	Weizmann	MMS and AME	1	96.7%
Gorelick et al. [22]	10	9	Weizmann	space-time shapes	1	97.8%
Weinland et al. [19]	10	9	Weizmann	silhouette	1	100%
Gu et al. [12]	11	12	IXMAS	3D human joint sequence	3D	94.4%
Weinland et al. [6]	11	10	IXMAS	MHV template	3D	93.3%
Lv et al. [7]	22	—	MoCap	3D human joint sequence	3D	92.1%
Davis et al. [23]	18	1	—	MEI and MHI	2	83.3%
Ahmad et al. [16]	7	11	KUGDB	optic flow and shape flow	3	88.3%
Natarajan et al. [17]	6	4	—	optic flow and shape flow	1	78.85%
Weinland et al. [8]	11	10	IXMAS	silhouette	4	81.3%
Yan et al. [9]	11	12	IXMAS	STVs	4	78%
Liu et al. [10]	13	12	IXMAS	STVs and spin-images	4	78.5%
our approach	11	12	IXMAS	3D key pose sequence	4	93%

Table 6 Comparison of the proposed approach with previous researches (IXMAS dataset)

	cam 1	cam 2	cam 3	cam 4	cam 2, cam 4	cam 1, cam 2 cam 3	cam 1, cam 2 cam 3, cam 4
Weinland et al. [8]	65.4%	70.0%	54.3%	66.0%	81.3%	—	81.3%
Yan et al. [9]	72%	53%	68%	63%	71%	60%	78%
Liu et al. [10]	73.46%	72.74%	69.62%	70.94%	—	—	78.5%
our approach	78%	78%	67%	78%	88%	87%	93%

of the actor by considering actions as 3D objects (x, y, t). All of the three mentioned algorithms adopted appearance-based action representation. In this paper, we use human pose-based action representation. First, 2D image action sequence is represented by 3D key pose sequence. Then, the pose property and dynamic property of 3D key pose sequence are modeled, respectively, and fused to recognize actions. Compared with the mentioned researches, the approach proposed in this paper yields better results.

6 Conclusion

This paper presents a new framework for representing and recognizing actions captured from arbitrary views. The main contributions are the view-independent action representation using 3D-key-pose set and multiple models fusion for action recognition for arbitrary view observations. Recovering 3D pose from arbitrary view action sequence is very difficult, due to the loss of depth information and self-occlusion. In this paper, 3D-key-pose set is used to obtain the 3D key pose sequence of arbitrary view action sequence. Exemplar-based embedding and probability of unique key pose are applied to model the pose property of 3D key pose sequence. Moreover, complementary dynamic feature is extracted from 3D key pose sequence to model these actions that share the same poses but have different dynamic feature. Finally, these action models are fused to recognize action sequence from a single or few cameras. Experimental results prove that the proposed method is effective. In addition, when observations from multiple cameras are fused, the recognition performance of proposed algorithm is comparable with those of 3D actions.

However, near perfect silhouette is needed for matching with 3D key pose. In some real environments, it is hard to obtain clean silhouette. Future work plans include improving silhouette extraction performance and recognizing actions from arbitrary viewpoints in real complex environments.

Acknowledgements This work was supported by the National Basic Research Program of China (973 program) under Grant No. 2007CB311004, and the National High Technology Research and Development Program of China (863 program) under Grant No. 2006AA01Z115.

References

1. Yilmaz A, Shah M. Matching actions in presence of camera motion. Computer Vision and Image Understanding, 2006, 104(2–3): 221–231
2. Poppe R. Vision-based human motion analysis: An overview. Computer Vision and Image Understanding, 2007, 108(1–2): 4–18
3. Shen Y, Ashraf N, Foroosh H. Action recognition based on homography constraints. In: Proceedings of International Conference on Pattern Recognition. 2008, 1–4
4. Souvenir R, Babbs J. Learning the viewpoint manifold for action recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2008, 1–7
5. Ahmad M, Lee S. HMM-based human action recognition using multiview image sequences. In: Proceedings of International Conference on Pattern Recognition. 2006, 1: 263–266
6. Weinland D, Ronfard R, Boyer E. Free viewpoint action recognition using motion history volumes. Computer Vision and Image Understanding, 2006, 104(2): 249–257
7. Lv F, Nevatia R. Recognition and segmentation of 3-D human action using HMM and multi-class AdaBoost. In: Proceedings of European Conference on Computer Vision. 2006, 359–372
8. Weinland D, Boyer E, Ronfard R. Action recognition from arbitrary views using 3D exemplars. In: Proceedings of IEEE International Conference on Computer Vision. 2007, 1–7
9. Yan P, Khan S M, Shah M. Learning 4D action feature models for arbitrary view action recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2008, 1–7
10. Liu J, Ali S, Shah M. Recognizing human actions using multiple features. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2008, 1–8
11. Johansson G. Visual motion perception. Scientific American, 1975, 232(6): 76–88
12. Gu J, Ding X, Wang S, Wu Y. Action and gait recognition from recovered 3-D human joints. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 2010, 40(4): 1021–1033
13. Cheung K M G, Baker S, Kanade T. Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2003, 1: 77–84
14. Parameswaran V, Chellappa R. View invariance for human action recognition. International Journal of Computer Vision, 2006, 66(1): 83–101
15. Gritai A, Sheikh Y, Shah M. On the use of anthropometry in the invariant analysis of human actions. In: Proceedings of International Conference on Pattern Recognition. 2004, 2: 923–926
16. Ahmad M, Lee S W. Human action recognition using shape and CLG-motion flow from multi-view image sequences. Pattern Recognition, 2008, 41(7): 2237–2252
17. Natarajan P, Nevatia R. View and scale invariant action recognition using multiview shape-flow models. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2008, 1–8
18. Lv F, Nevatia R. Single view human action recognition using key pose matching and Viterbi path searching. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2007, 1–8
19. Weinland D, Boyer E. Action recognition using exemplar-based embedding. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2008, 1–7
20. Rabiner L R. A tutorial on hidden Markov model and selected applications in speech recognition. Proceedings of the IEEE, 1989, 77(2): 257–286
21. Wang L, Suter D. Informative shape representations for human action recognition. In: Proceedings of International Conference on

Pattern Recognition. 2006, 1266–1269

22. Gorelick L, Blank M, Shechtman E, Irani M, Basri R. Actions as space-time shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(12): 2247–2253

23. Davis J W, Bobick A F. The representation and recognition of human movement using temporal templates. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 1997, 928–934