




Technical Correspondence

Recognition and Detection of Two-Person Interactive Actions Using Automatically Selected Skeleton Features

Huimin Wu , Jie Shao , Xing Xu , Yanli Ji, Fumin Shen, and Heng Tao Shen

Abstract—Recognition and detection of interactive actions performed by multiple persons have a wide range of real-world applications. Existing studies on the human activity analysis focus mainly on classifying video clips of simple actions performed by a single person, whereas the problem of understanding complex human activities with causal relationships between two people has not been sufficiently addressed yet. In this paper, we employ systematically organized skeleton features enhanced with directional features, and utilize sparse-group lasso to automatically choose discriminative factors that help in dealing with interactive action recognition and real-time detection tasks. Experiments on two person interaction datasets demonstrate the superiority of our approach to the state-of-the-art methods.

Index Terms—Interactive action, real-time detection, skeleton features.

I. INTRODUCTION

Human action analysis has many applications in surveillance, human computer interaction, gaming, etc., [1]. Most previous researches in this field focused on understanding video clips of simple actions performed by a single person, e.g., “running” or “waving” [2], [3]. However, in real-world scenarios actions are often performed by multiple persons, e.g., “pushing” or “hugging” [4]–[6]. Recognition of this kind of complex human activities will be necessary for a number of demands. Besides offline action recognition, real-time action detection usually meets actual requirements better, such as automatic detection of violent activities in smart surveillance systems. However, only a few research efforts have been made to explicitly address the problem of interactive action detection.

In terms of feature extraction, broadly two types of techniques were developed for understanding human actions: *space-time feature extraction approaches* [7], [8] and *biologically plausible approaches* [9], [10]. Local space-time features are generalized mainly from low-level image features. For example, 3D-SIFT [7] is developed from SIFT, and HOG3D [11] from the histogram of oriented gradients

(HOG). Biologically plausible approaches tend to be perception-based and more intuitive. For example, Xia *et al.* [12] used histograms of three-dimensional (3-D) joint locations (HOJ3D) to represent postures and built visual words by employing linear discriminant analysis on HOJ3D, and Yun *et al.* [5] extracted multiple body-pose features by using joint position data captured by Microsoft Kinect.

As joint position trajectories are adequate for humans to recognize actions according to [9], human joint sequence is an effective representation for structured motion [13]. In particular, the development of depth sensors, such as Microsoft Kinect, makes available 3-D joint position sequences [14], which enable action recognition and detection.

Some joints are discriminative for recognizing an action while others are not, and this aspect is ignored by many previous works. Zhu *et al.* [15] proposed to perform automatic feature learning by using regularization scheme to learn the co-occurrence of joints. However, this kind of regularization does not differentiate joints directly, and as a result of which feature selection will be less informative. In this paper, we employ *static and dynamic skeleton features* in the form of a sequence of tracked human joints inferred by using RGBD (i.e., color plus depth) sensor, which has been proved experimentally to be simply and effective [5], [16]. More importantly, regularization on weight parameters, properly grouped, helps in choosing discriminative factors for recognition and detection of interactive actions.

Our work makes the following technical contributions.

- 1) We propose to incorporate additional *directional features* to complement static and dynamic features that contain only local distance information (see Sections III-A and III-B). Incorporation of global directional information can provide more representative features.
- 2) For accurate action recognition, after extracting representative features, we adopt sparse-group lasso on properly organized features for *automatic feature selection* (see Section III-C).
- 3) In addition to interactive action recognition, we address a new *real-time interactive action detection* problem by simulating real-life detection process (see Section IV).
- 4) We conduct experiments on two skeleton-based human interaction datasets, SBU [5] and CR-UESTC [6], both captured by a stationary Microsoft Kinect, and compare the results with those of the state-of-the-art methods. The comparison demonstrates the efficacy of our approach by virtue of its more accurate results (see Section V).

II. RELATED WORK

Skeletons, estimated from images/videos or captured by depth sensors, are widely used for action recognition. For example, Li *et al.* [17] sampled 3-D points and modeled the dynamic evolution of actions for action recognition. However, general techniques for simple actions

Manuscript received May 15, 2017; revised August 22, 2017; accepted October 25, 2017. Date of publication December 4, 2017; date of current version May 15, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61672133, Grant 61602089, Grant 61673088, Grant 61502081, and Grant 61632007, and in part by the Fundamental Research Funds for the Central Universities under Grant ZYGX2015J058 and Grant ZYGX2014Z007. This paper was recommended by Associate Editor Dr. Xiaogang Hu. (Corresponding author: Jie Shao.)

H. Wu, J. Shao, X. Xu, F. Shen, and H. T. Shen are with the Center for Future Media and the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: wuhuimin@std.uestc.edu.cn; shaojie@uestc.edu.cn; xing.xu@uestc.edu.cn; fshen@uestc.edu.cn; shenhengtao@uestc.edu.cn).

Y. Ji is with the School of Automation Engineering and the Center for Future Media, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: yanliji@uestc.edu.cn).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/THMS.2017.2776211

performed by a single person do not normally work well for interaction recognition, because mutual occlusions and overlaps may lead to large noise on skeleton estimation [6].

Feature extraction on skeleton data: For interaction recognition, the basic scheme for feature extraction is using original (filtered in temporal domain for noise robustness) 3-D positions of joints [15], [18]. Using the positions of joints directly seems less reliable, because semantically similar actions may not guarantee numerical similarity [19]. For example, the actions of both the pairs of persons, shaking hands at different places, will be recognized as “hand shaking.” However, the positions of these two pairs of persons can be very different. In other words, the specific positions of joints do not recognize the action, but the pose formed by joints does. Instead of using original position features, some studies extract relative joint features for more effective representation. Alazrai *et al.* [20] represented interactions with motion-pose geometric descriptors, and showed that skeleton-based representation outperforms local space-time feature based representation. Following the feature extraction approach in [19], Yun *et al.* evaluated a number of relative distance features for two-person interaction recognition on SBU dataset [5], and found that joint features of joint distance and joint motion perform best. Ji *et al.* [6] modified the feature extraction approach in [5] and [16] and proposed a contrastive feature distribution model (CFDM) for interaction recognition. In this paper, we adopt the *static and dynamic features* proposed in [5], but additionally propose to incorporate *directional features* as direct and informative representation for interactive action recognition on skeleton data.

Real-time action detection with skeleton data: Only a few studies addressed interactive action detection on skeleton data. Sung *et al.* [21] built a two-layered Markov model, using Euclidean coordinates and orientation matrix of each joint. However, their setting of “detection” refers simply to classification of presegmented short sequences around the peak of interest, which is inconsistent with the usual notion of “detection” in real-life applications. Normally, “detection” also needs to distinguish actions of interest from random actions. In this paper, the action “detection” task is defined based on a sliding-window scheme. By solving interactive action detection with promising results, we prove that our solution can be applied to both action recognition and action detection tasks.

In summary, interactive action recognition and detection rely on informative feature extraction. It is desirable to ignore redundant features and identify discriminative factors to represent interactions in a way that improves the accuracy. To this end, a novel automatic feature selection approach is proposed here.

III. RECOGNITION OF INTERACTIVE ACTIONS AND JOINT FEATURE SELECTION

Static and dynamic skeleton features are used in this paper, because of their proven simplicity and experimental efficiency [5], which will be discussed in Section III-A. To obtain more descriptive and distinguishing features, we propose to additionally incorporate directional information in Section III-B. Then, to leverage the contribution of each joint and each person to a specific action, discriminative features are automatically selected by applying sparse-group lasso [22], which will be explained in Section III-C.

In the following, we use p_1 and p_2 to denote two participants for an interactive action. Supposing that the positions of N_J joints are captured for each person, we use J_1, J_2, \dots, J_{N_J} to denote joints and index them with notations i, k . Generally, we use $a, b \in \{1, 2\}$ to index participants and $j, l \in \{1, 2, \dots, N_v\}$ to index frames, where N_v denotes the number of frames. We define $l(J_i; p_a, f_j)$ (or $l(J_k; p_b, f_l)$)

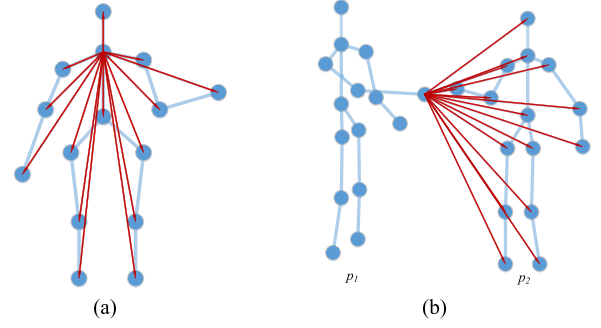


Fig. 1. Example of static feature of a joint, in which blue solid circles represent joints, blue lines joint connections, and red arrows static feature vectors of the joint. (a) Static feature for joint “neck” related to the same participant. (b) Static feature for joint “right hand” related to the other participant.

as the position of some joint J_i (J_k), which belongs to participant p_a (p_b) in frame f_j (f_l).

A. Static and Dynamic Skeleton Features

Distance between joint J_i of participant p_a in frame f_j with position $l(J_i; p_a, f_j)$ and joint J_k of participant p_b in frame f_l with position $l(J_k; p_b, f_l)$ is defined as follows:

$$d(J_i, J_k; p_a, p_b, f_j, f_l) = \|l(J_i; p_a, f_j), l(J_k; p_b, f_l)\| \quad (1)$$

where $\|\cdot\|$ denotes Euclidean distance.¹

Static joint features correspond to the poses of participants, which are captured by the distances between joints in the same frame. Dynamic joint features correspond to the motion of participants, which are represented by the distances between joints in different frames. As the distances of joints in two far-apart frames are useless for action recognition (e.g., the distance between elbow joint in the first frame and torso joint in the last frame is not helpful for recognizing the action “punching”), we follow the work of [5] and extract dynamic features with frames bounded to a window.

Unless otherwise stated, we use the features of participant p_1 for description. Within a window of length 3 (default value in this paper, which suits well as informative and compact representation for test videos) represented as a frame index set $\mathcal{W} = \{\eta, \eta + 1, \eta + 2\}$, static feature vector of a joint J_i of participant p_1 relating to joints of the same participant is denoted by $\mathcal{SF}_S(J_i; p_1, \mathcal{W})$ with $(N_J - 1) \times 3$ elements in total, and the element of the vector is $d(J_i, J_k; p_1, p_1, f_j, f_j)$, where $j \in \mathcal{W}$ and $J_i \neq J_k$. As exemplified in Fig. 1(a), each red arrow corresponds to an element of vector $\mathcal{SF}_S(J_i; p_1, \mathcal{W})$. Static feature vector of a joint J_i relating to joints of the other participant is represented by $\mathcal{SF}_O(J_i; p_1, \mathcal{W})$ with $N_J \times 3$ elements in total, and the element of the vector is $d(J_i, J_k; p_1, p_2, f_j, f_j)$, where $j \in \mathcal{W}$. Each red arrow in Fig. 1(b) corresponds to an element of vector $\mathcal{SF}_O(J_i; p_1, \mathcal{W})$. Static feature vector of a joint J_i is represented as $\mathcal{SF}(J_i; p_1, \mathcal{W})$ concatenated by $\mathcal{SF}_S(J_i; p_1, \mathcal{W})$ and $\mathcal{SF}_O(J_i; p_1, \mathcal{W})$, with a length of $6N_J - 3$.

Similarly within \mathcal{W} , dynamic feature vector of a joint J_i relating to joints of the same participant is $\mathcal{DF}_S(J_i; p_1, \mathcal{W})$ (with $N_J \times 6$ elements), whose element is $d(J_i, J_k; p_1, p_1, f_j, f_l)$, where $j, l \in \mathcal{W}$ and $j < l$ (or $j > l$). For example, as shown in Fig. 2(a), each red arrow corresponds to $d(J_i, J_k; p_1, p_1, f_1, f_2)$ and each green arrow corresponds to $d(J_i, J_k; p_1, p_1, f_1, f_3)$ where J_i represents “right hand” and J_k represents some joint that belongs to

¹ $\|(x_1, y_1, z_1), (x_2, y_2, z_2)\| = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$.

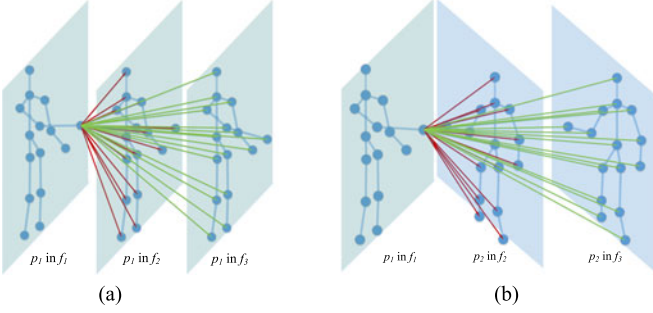


Fig. 2. Example of dynamic feature of a joint, in which blue solid circles represent joints, blue lines joint connections, and red/green arrows dynamic feature vectors of the joint. (a) Dynamic feature for joint “right hand” related to the same participant. (b) Dynamic feature for joint “right hand” related to the other participant.

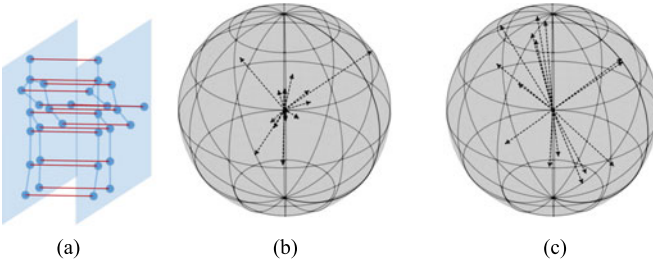


Fig. 3. Example showing the distribution of movement vectors of two participants in a frame. (a) Movement vectors of each joint for left participant. (b) Directional vectors of left participant. (c) Directional vectors of right participant.

the same participant (for clarity of illustration, the arrows corresponding to $d(J_i, J_k; p_1, p_1, f_2, f_3)$ are not shown). Dynamic feature vector of a joint J_i relating to joints of the other participant is $\mathcal{DF}_O(J_i; p_1, \mathcal{W})$ (with $N_J \times 6$ elements), whose element is $d(J_i, J_k; p_1, p_2, f_j, f_l)$, where $j, l \in \mathcal{W}$ and $j < l$. As exemplified in Fig. 2(b), each red arrow corresponds to $d(J_i, J_k; p_1, p_2, f_1, f_2)$ and each green arrow corresponds to $d(J_i, J_k; p_1, p_2, f_1, f_3)$ where J_i represents “right hand” and J_k represents some joint that belongs to the other participant (for clarity of illustration, the arrows corresponding to $d(J_i, J_k; p_1, p_2, f_2, f_3)$ are not shown). Dynamic feature vector of a joint J_i is represented as $\mathcal{DF}(J_i; p_1, \mathcal{W})$ concatenated by $\mathcal{DF}_S(J_i; p_1, \mathcal{W})$ and $\mathcal{DF}_O(J_i; p_1, \mathcal{W})$, with a length of $12N_J$.

B. Directional Features

Static and dynamic features contain information on only distance and its variations, but direction of joint movement is also essential for interactive action understanding. Given the position of each joint in three dimensions, the movement vector of each joint *from each frame to next frame* can be calculated. Here, the movement vector of each joint in two successive frames is no longer considered scalar. Since the magnitude of a movement vector is already included in Section III-A (dynamic feature vector of a joint J_i relates to joints of the same participant), we consider each movement vector with a unit length, and focus only on its directional information in three dimensions. Moreover, instead of using coarse and naive representation of movement vector direction based on each joint, we employ the concept of entropy to represent the directional features based on each participant, as an additional feature to complement static and dynamic features.

For each participant, the movement vector of each joint can be plotted in three dimensions as shown in Fig. 3 (dotted arrows). With basic

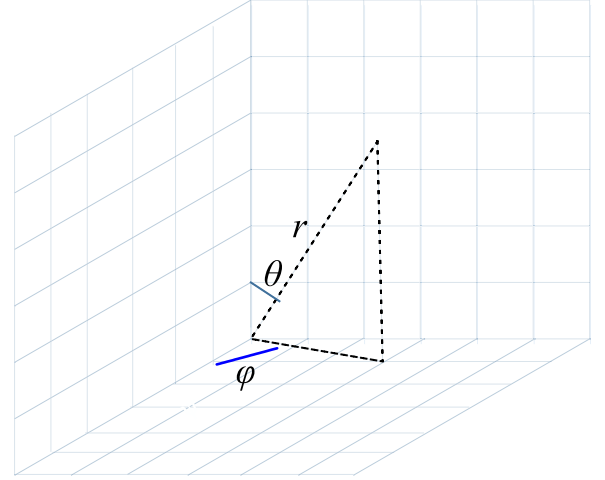


Fig. 4. Position of a point in spherical coordinate system can be specified by three values: r , θ , and ϕ , where r represents *radial distance* between the point and the origin, θ represents *polar angle* measured as an angle from zenith direction, and ϕ represents *azimuth angle* measured as counterclockwise angle from X -axis to its orthogonal projection on XOY plane.

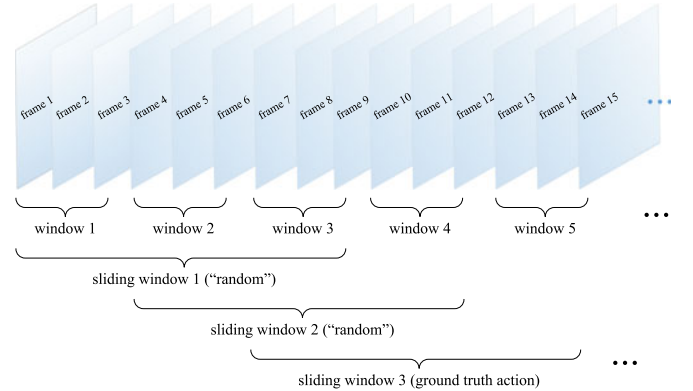


Fig. 5. Sliding-window based action detection model and ground truth.

knowledge in geometry, it is easy to understand that each movement vector can be mapped to a point in a sphere with unit radius, which can be specified in terms of three values r , θ , and ψ , as shown in Fig. 4.

We discretize these vectors for convenience in manipulating such vectors. Splitting the sphere into several zones, just as the earth can be split, based on longitude and latitude (illustrated as the meshes in Fig. 3), we index each zone and only need to calculate the zone index where a vector locates, instead of using the precise position of the vector. Experimentally, we split the sphere along every 30° longitude and every 30° latitude² into $(180^\circ/30^\circ) \times (360^\circ/30^\circ) = 72$ zones.

If Z_i denotes a zone, then the entropy of the movement vector direction of all joints for participant p_1 in frame f_j with $j \in \mathcal{W}$ is defined as

$$H(p_1; f_j, \mathcal{W}) = - \sum_i P(Z_i; f_j, \mathcal{W}) \log P(Z_i; f_j, \mathcal{W}) \quad (2)$$

where $P(Z_i; f_j, \mathcal{W})$ is the probability of directional vectors mapped to Z_i (calculated as the frequency of vectors located at zone Z_i). We use $H(p_1; \mathcal{W}) = [H(p_1; f_j, \mathcal{W}), H(p_1; f_{j+1}, \mathcal{W}), H(p_1; f_{j+2}, \mathcal{W})]$ to denote the directional features relating to participant p_1 in win-

²This value is set to balance between nontrivialness and noncoarseness.

TABLE I
ACCURACY COMPARISON IN SBU AND CR-UESTC DATASETS

Approach \ Dataset	SBU	CR-UESTC
Joint features+SVM [5]	0.803	-
CFDM [6]	0.894	0.876
Hierarchical RNN [18]	0.804	-
Deep LSTM+Co-occurrence+In-depth dropout [15]	0.904	-
Our approach	0.910	0.892

dow \mathcal{W} . Similarly, the directional features relating to participant p_2 in window \mathcal{W} can be denoted as $H(p_2; \mathcal{W})$.

C. Choosing Discriminative Factors

Arm joints are more important than lower body joints for recognizing “hugging” action, just as leg joints are more important than upper body joints for recognizing “kicking” action. With this in view, we need to leverage the contribution of *different joints* to different actions. In addition, some interactive actions are completed mainly by one participant (the “actor”) followed by the response of the other participant (the “acceptor”) [23], and this requires weighting of the contributions of joints belonging to *different participants*. Besides, considering that the importance of distance-related features, relative to direction-related features, can be different for different actions, we also need to distinguish the contribution between *different feature types*.

We employ sparse group lasso [22] to discriminate the contributions of *different joints*, *different participants*, and *different feature types* by exploring sparsity of groups. Each line of feature matrix X , denoted as x , is a feature vector of a video. As the length of each video varies, we only use several windows around the peak of the action to conduct interaction recognition and process the whole sequence in interaction detection. We extract features of N_W windows, and organize features for each window \mathcal{W} of video v as $x_{\mathcal{W}} = [\mathcal{SF}(J_1; p_1, \mathcal{W}), \mathcal{DF}(J_1; p_1, \mathcal{W}), \dots, \mathcal{SF}(J_{N_J}; p_1, \mathcal{W}), \mathcal{DF}(J_{N_J}; p_1, \mathcal{W}), \dots, \mathcal{DF}(J_{N_J}; p_2, \mathcal{W}), H(p_1; \mathcal{W}), H(p_2; \mathcal{W})]$. Feature matrix X is concatenated by $x_{\mathcal{W}}$, i.e., $x = [x_1, x_2, \dots, x_{N_W}]$. Training ground truth Y is defined by using one-versus-all strategy, i.e., by using a zero vector with a “1” in the position that indicates the action.

We solve the parameter matrix W on a dataset with N_V videos defined in sparse-group lasso problem by

$$\min_W \frac{1}{2N_V} \|Y - \sum_{g=1}^m X^{(g)} W^{(g)}\|_2^2 + \lambda_1 \sum_{g=1}^m \|W^{(g)}\|_2 + \lambda_2 \|W\|_1 \quad (3)$$

where $X^{(g)}$ is the submatrix of X , which comprises columns of X belonging to group g , and $W^{(g)}$ is the submatrix of W , which comprises rows of W corresponding to $X^{(g)}$. Each group corresponds to nondirectional features $\mathcal{SF}(J_i; p_a, \mathcal{W})$ and $\mathcal{DF}(J_i; p_a, \mathcal{W})$ with a length of $18N_J - 3$, or directional features $H(p; \mathcal{W})$ with a length of $6 = 3(\text{frames}) \times 2(\text{participants})$. m denotes the number of groups. λ_1 and λ_2 are the two tuning parameters.

Using one-versus-all strategy, it is easy to understand that $W^{(g)}$ and Y are with the same column size as the action class number. We focus on one column c of $W^{(g)}$ and Y represented as $W_c^{(g)}$ and Y_c , respectively, in order to solve the sparse group lasso problem in (3). Generally, we represent feature matrix $X^{(g)}$ as $X^{(g)} = [\chi_1, \chi_2, \dots, \chi_{n_g}]$, and $W^{(g)}$ as $W^{(g)} = [\omega_1, \omega_2, \dots, \omega_{n_g}]$. The subgradient equations with respect to each item of $W_c^{(g)}$ are

$$-\chi_j^T (r - \sum_j \chi_j \omega_j) + \lambda_1 s_j + \lambda_2 t_j = 0 \quad (4)$$

TABLE II
EFFECT OF DIRECTIONAL FEATURES BY COMPARING ACCURACY WITH AND WITHOUT DIRECTIONAL FEATURES

Approach \ Dataset	SBU	CR-UESTC
With directional features	0.910	0.892
Without directional features	0.882	0.888

where $r = y_c - \sum_{k \neq c} \chi_k \omega_k$ and

$$s_j = \begin{cases} \omega_j / \|W_c^{(g)}\|, & \omega_j \neq 0 \\ \in \{s : \|s\| \leq 1\}, & \text{otherwise} \end{cases}$$

$$t_j = \begin{cases} \text{sign}(\omega_j), & \omega_j \neq 0 \\ \in [-1, 1], & \text{otherwise.} \end{cases}$$

Detailed derivations can be found in [22].

Although (3) looks similar to co-occurrence regularization in [15], our work is fundamentally different from [15] in which LSTM neurons are divided into groups. We organize skeleton features into a group in an intuitive and interpretable way, to distinguish the contributions of *different feature types* with regard to *different joints* belonging to *different participants*.

IV. SLIDING-WINDOW BASED DETECTION OF INTERACTIVE ACTIONS

In [5], action “detection” is achieved by recognizing actions from a window around the peak of the action of interest, which is essentially a recognition problem because the peak of the action of interest is actually unknown in advance in an online detection settings. In this paper, we define the real-time action detection problem in a more practical manner, and solve it with a sliding-window based algorithm. As shown in Fig. 5, we group successive N_{GT} (set to be 3 in this paper) windows into a sliding window. The sliding window around the peak of interest is labeled as the ground truth action and the other sliding windows as “random.” In this way, we get the sliding-window level ground truth, and the features extracted from a sliding window are used for action classification. The windows covered by sliding windows are labeled as ground truth action. In short, for action recognition task, we have video-level ground truth, and for action detection task, we have sliding-window ground truth and window-level ground truth. To evaluate the performance of an action detection algorithm, three aspects need to be considered, which are as follows:

- 1) a few sliding windows should be mislabeled;
- 2) a few windows belonging to an action should be mislabeled as other actions;
- 3) a few “random” windows should be detected as ground truth action.

These three aspects are evaluated by *accuracy*, *precision*, and *recall*, respectively.

V. EXPERIMENTAL EVALUATION

A. Datasets and Evaluation Metrics

We conduct our experiments on two datasets depicting two-person interactions, SBU [5] and CR-UESTC [6]. Both datasets contain multiple sets, and each set contains a pair of participants who performed all actions. The SBU dataset contains 21 pairs of participants and 8 actions, and the CR-UESTC dataset contains 25 pairs of participants and 10 actions. Interactive action recognition performance is evaluated from

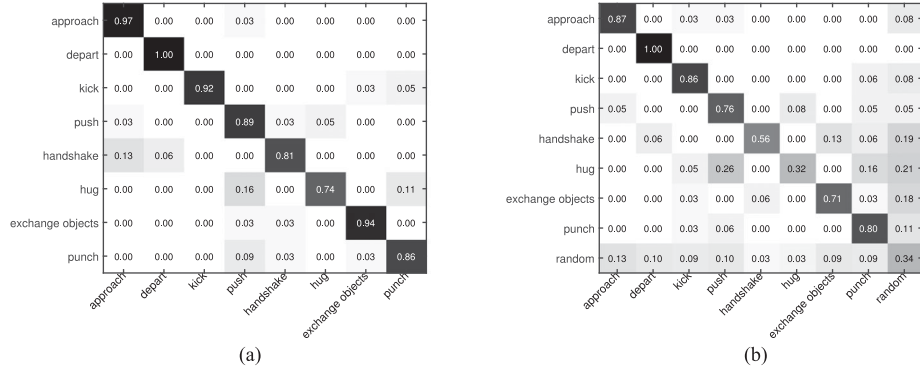


Fig. 6. Confusion matrices of different tasks in SBU dataset. (a) Action recognition. (b) Action detection.

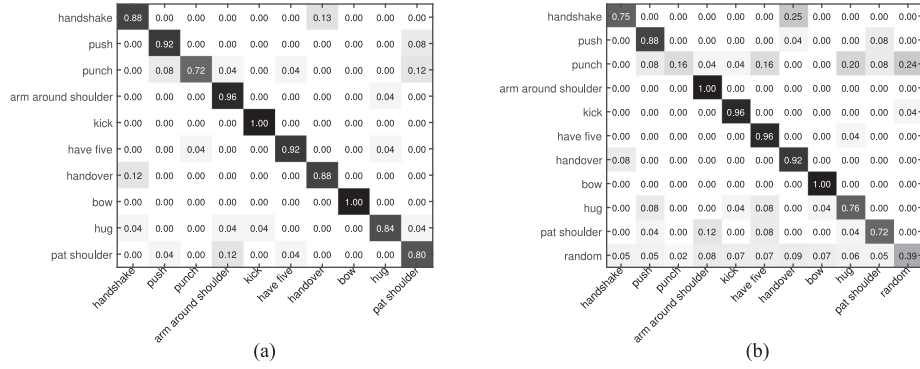


Fig. 7. Confusion matrices of different tasks in CR-UESTC dataset. (a) Action recognition. (b) Action detection.

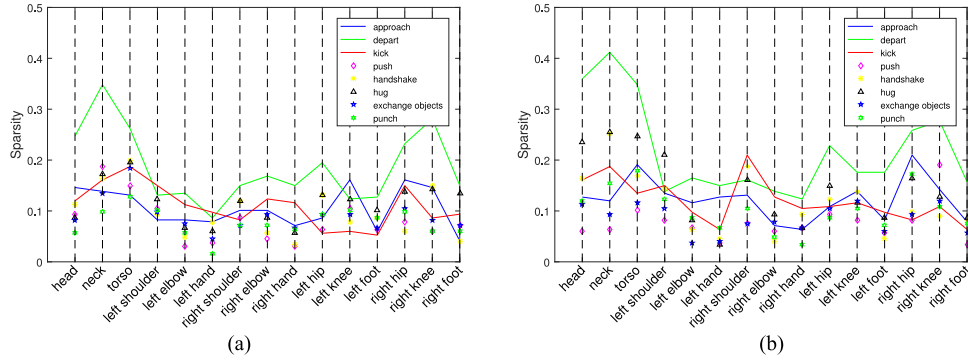


Fig. 8. Sparsity on SBU dataset. (a) Sparsity related to joints of left participants for action recognition task on SBU dataset. (b) Sparsity related to joints of right participants for action recognition task.

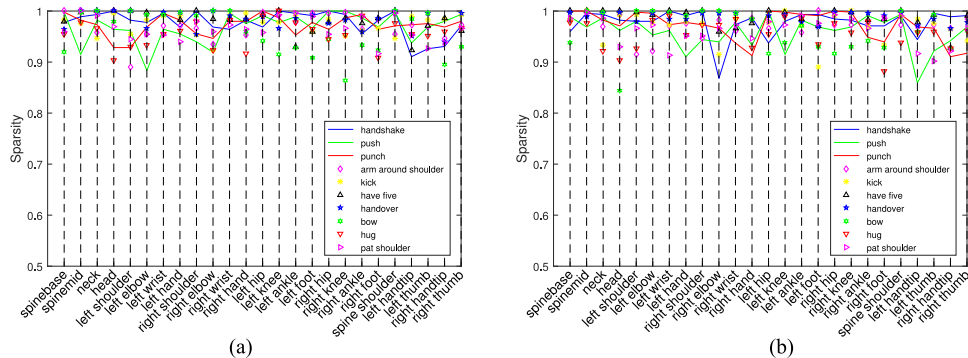


Fig. 9. Sparsity on CR-UESTC dataset. (a) Sparsity related to joints of left participants for action recognition task on CR-UESTC dataset. (b) Sparsity related to joints of right participants for action recognition task.

TABLE III
PERFORMANCE OF OUR ACTION DETECTION APPROACH IN SBU DATASET

Approach Performance	Random assignments	Without directional features			With directional features		
		best <i>Accuracy</i> ₂	best <i>Precision</i>	best <i>Recall</i>	best <i>Accuracy</i> ₂	best <i>Precision</i>	best <i>Recall</i>
<i>Accuracy</i> ₂	0.118	0.404	0.392	0.386	0.422	0.402	0.405
<i>Precision</i>	0.220	0.496	0.508	0.491	0.503	0.517	0.512
<i>Recall</i>	0.260	0.851	0.830	0.859	0.849	0.860	0.875

TABLE IV
PERFORMANCE OF OUR ACTION DETECTION APPROACH IN CR-UESTC DATASET

Approach Performance	Random assignments	Without directional features			With directional features		
		best <i>Accuracy</i> ₂	best <i>Precision</i>	best <i>Recall</i>	best <i>Accuracy</i> ₂	best <i>Precision</i>	best <i>Recall</i>
<i>Accuracy</i> ₂	0.089	0.407	0.406	0.399	0.410	0.409	0.403
<i>Precision</i>	0.195	0.324	0.329	0.323	0.325	0.335	0.328
<i>Recall</i>	0.307	0.854	0.857	0.862	0.858	0.867	0.869

video level by accuracy_1 , and interactive action detection performance is measured by accuracy_2 from sliding-window level, and precision and recall from window level. Supposing among A actions in test set, TPA is the set that is correctly labeled, video-level performance is evaluated by $\text{accuracy}_1 = \frac{\# \text{TPA}}{A}$. Sliding-window level evaluation metric is defined as $\text{accuracy}_2 = \frac{\# \text{TPSW}}{\# \text{SW}}$, where TPSW and SW denote the sets of sliding windows correctly labeled and sliding windows in test set, respectively. For a video v , let TPW_v , PW_v , and TW_v denote the sets of windows correctly detected as the ground truth action, windows detected as the ground truth action and windows labeled as the ground truth action. Window-level performance is evaluated as: $\text{precision} = \sum_v \frac{\# \text{TPW}_v}{\# \text{PW}_v}$, and $\text{recall} = \sum_v \frac{\# \text{TPW}_v}{\# \text{TW}_v}$. These metrics are averaged over five-fold participant-independent cross validation.

B. Interaction Recognition Performance

Comparison with the state-of-the-art methods: We compare the performance of action recognition obtained by our approach with the approaches proposed in [5], [6], [15], and [18] in Table I. It can be seen that our approach performs better than the existing best performance reported on each dataset. Confusion matrices of our approach for action recognition tasks are shown in Figs. 6(a) and 7(a). In particular, compared with the deep model [15], our model is a lightweight one (without dependence on hardware such as GPUs), and its results are easy to interpret because the contribution of each joint to the recognition of an action can be reflected in the model, as will be discussed later.

Analysis on misclassification results: Some actions can be mislabeled, because they are naturally so similar that they are hard to distinguish even for human perception. For example, in SBU dataset [see Fig. 6(a)] action “hug” can sometimes be confused for “push,” and action “handshake” has a good chance of being mislabeled as “approach.” In CR-UESTC dataset [see Fig. 7(a)], “handshake” and “handover” are likely to be mislabeled one for the other, action “punch” has a good chance of being mislabeled as “pat shoulder,” and “pat shoulder” can sometimes be confused for “arm around shoulders.”

Effect of directional features: To evaluate the effect of directional information, we compare the performances of two action recognitions, one with directional features and the other without. The accuracies achieved by these two methods are shown in Table II from which it can be seen how the results underscore the need to use directional features for interaction representation.

Effect of automatic feature selection: To evaluate the effect of automatic feature selection in our approach, we introduce *sparsity* of parameters corresponding to each joint to reflect the contribution of the joint to an action. For each action, the sparsity related to a joint is obtained by calculating the ratio of zero parameters in the parameters

corresponding to the joint in learned W . In general, the more the contributions of a joint to an action, the lower would be the sparsity related to the joint. For example, for the action “push,” two most important joints for the left participant in SBU dataset (see Fig. 8) are “right hand” and “left elbow,” and for the right participant “left hand” and “right foot”. In CR-UESTC dataset (see Fig. 9), two most important joints for the left participant are “left elbow” and “spinemid,” and for the right participant “left handtip” and “left hand.” These results are consistent with our observations and intuitions.

C. Interaction Detection Performance

We compare the performance of our action detection approach with the averaged detection results of random assignments on both the two datasets. Averaged results over five-fold cross validation, in pursuit of each metric, are reported here. Tables III and IV demonstrate the superiority of our approach by showing that its performance is much better in terms of all evaluation metrics. Confusion matrices of our approach for interaction detection task are shown in Figs. 6(b) and 7(b).

From the foregoing results, it follows that, in terms of all metrics, interaction detection performs better with directional features than without. Also, the performance on both interaction recognition and detection tasks highlights the need for using directional features.

VI. CONCLUSION

In this paper, we studied the problem of interactive action recognition, using automatically selected joint features, which contain both distance information and directional information. We solved the real-time interactive action detection task by a sliding-window based technique. Our experimental results demonstrate the effectiveness of our detection algorithm and also confirm that our recognition approach outperforms the state-of-the-art methods.

REFERENCES

- [1] J. K. Aggarwal and M. S. Ryoo, “Human activity analysis: A review,” *ACM Comput. Surveys*, vol. 43, no. 3, 2011, Art. no. 16.
- [2] J. C. Niebles, H. Wang, and F. Li, “Unsupervised learning of human action categories using spatial-temporal words,” *Int. J. Comput. Vis.*, vol. 79, no. 3, pp. 299–318, 2008.
- [3] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, “Actions as space-time shapes,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2247–2253, Dec. 2007.
- [4] M. S. Ryoo and J. K. Aggarwal, “Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities,” in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, 2009, pp. 1593–1600.

- [5] K. Yun, J. Honorio, D. Chattopadhyay, T. L. Berg, and D. Samaras, "Two-person interaction detection using body-pose features and multiple instance learning," in *Proc. 2012 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2012, pp. 28–35.
- [6] Y. Ji, H. Cheng, Y. Zheng, and H. Li, "Learning contrastive feature distribution model for interaction recognition," *J. Visual Commun. Image Represent.*, vol. 33, pp. 340–349, 2015.
- [7] P. Scovanner, S. Ali, and M. Shah, "A 3-dimensional SIFT descriptor and its application to action recognition," in *Proc. 15th Int. Conf. Multimedia*, 2007, pp. 357–360.
- [8] I. Laptev and T. Lindeberg, "Space-time interest points," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, 2003, pp. 432–439.
- [9] G. Johansson, "Visual motion perception," *Sci. Amer.*, vol. 232, no. 6, pp. 76–88, 1975.
- [10] E. Yu and J. K. Aggarwal, "Human action recognition with extremities as semantic posture representation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1–8.
- [11] A. Kläser, M. Marszalek, and C. Schmid, "A spatio-temporal descriptor based on 3D-gradients," in *Proc. Brit. Mach. Vis. Conf.*, 2008, pp. 1–10.
- [12] L. Xia, C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3D joints," in *Proc. 2012 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2012, pp. 20–27.
- [13] J. Gu, X. Ding, S. Wang, and Y. Wu, "Action and gait recognition from recovered 3-D human joints," *IEEE Trans. Syst., Man, Cybern., B*, vol. 40, no. 4, pp. 1021–1033, Aug. 2010.
- [14] J. Shotton *et al.*, "Real-time human pose recognition in parts from single depth images," in *Proc. 24th IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1297–1304.
- [15] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie, "Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 3697–3704.
- [16] A. Yao, J. Gall, G. Fanelli, and L. J. V. Gool, "Does human action recognition benefit from pose estimation?" in *Proc. Brit. Mach. Vis. Conf.*, 2011, pp. 1–11.
- [17] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3D points," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 9–14.
- [18] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1110–1118.
- [19] M. Müller, T. Röder, and M. Clausen, "Efficient content-based retrieval of motion capture data," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 677–685, 2005.
- [20] R. Alazrai, Y. Mowafi, and C. S. G. Lee, "Anatomical-plane-based representation for human-human interactions analysis," *Pattern Recognit.*, vol. 48, no. 8, pp. 2346–2363, 2015.
- [21] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Human activity detection from RGBD images," in *Proc. Plan, Activity, Intent Recognit., Papers 2011 AAAI Workshop*, 2011, pp. 47–55.
- [22] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A sparse-group lasso," *J. Comput. Graph. Stat.*, vol. 22, no. 2, pp. 231–245, 2013.
- [23] Y. Ji, A. Shimada, H. Nagahara, and R. Taniguchi, "Contribution estimation of participants for human interaction recognition," *IEEJ Trans. Elect. Electron. Eng.*, vol. 8, no. 3, pp. 269–276, 2013.