Interactive Action Recognition using Relative Geometry Between Body Parts Across Multiple People

Swami Sankaranarayanan and Kota Hara

Center for Automation Research, University of Maryland, College Park, MD 20742

{swamiviv,kotahara}@umiacs.umd.edu

Abstract

In this work, we propose

1. Introduction

Human action recognition has been a major problem in computer vision community for many years. There are many applications of the human action recognition such as surveillance, health-care and human computer interaction. Most of the previous works focus on action recognition of a single person. The action classes considered in this scenario is limited to a simple action such as walking, running and jumping.

On the other hand, there are several works which deal with action classes involved with multiple people. These action classes are more complex to recognize than the single-person actions due to the complexity of the actions. The actions are defined not only by the motion of one person but also their interactive motions. We call these multiple-people actions as interactive actions.

The examples of interactive actions we consider in this work are exchanging objects and greetings between two people. For exchanging objects, we consider different types of the objects such as cards, a ball and a chair. Depending on the object being exchanged, their interactive movement come out in a different form. For greetings, we consider interactive actions such as shaking hands and bowing. Our goal is to recognize these different types of interactive actions from the 3d skeleton information.

Action recognition from the body skeleton information has been actively studied due to the advance of motion capture systems such as Kinect. Typically the skeleton information is given in the form of 3d locations of a set of body joints. Many action recognition systems directly use these information as the input or convert the location information into a set of joint angles and then use them as the input. The focus of most of the previous research has been on the recognition part.

[1] proposed a new skeleton representation based on relative geometry across body parts and achieve the state of the art results on several standard single-person action recognition datasets. They compute the translation and rotation between each pair of body parts and map it to the tangent space to obtain a feature vector. The benefit of their approach compared to the previously adopted skeleton representation is that their representation can directly capture the relationship between body parts that are not directly connected each other. Thus, it is more capable of recognizing actions in which the relationship between non-connected body parts, such as left arm and right arm, is more important.

In this work, we extend [1] to classify actions performed by two people. In our skeleton representation, in addition to the relative geometry across body parts within one person, we also consider relative geometry between body parts across two people. We believe that this representation enables us to capture two people's interaction well. For instance, our representation can captures the geometric relationship between a left arm of a person A and a right arm of a person B, which might be helpful to recognize the 'shaking hands' actions.

For the recognition part, we develop an algorithm to automatically detect frames where actions are taking place. For each sequence, we computer the proposed skeleton features at each frame in the detected periods of the frames. Then we sample a fixed number of features with the constant interval by a simple interpolation method. This way we obtain the same number of features from each detected period. We concatenate these features and use it as an input to multiclass classifiers.

The paper is organized as follows. In section 2, we show some related work. Section 3 explains our action localization method. Then we describe, in section 4, the proposed skeleton representation at each frame. Section 5 presents the interpolation method to obtain a final feature vector as well as the classification methods we propose. The experimental results are shown in section 6, followed by the conclusion in section 7.

2. Related Work

3. Action Localization

4. Skeleton Representation

The output from the Kinect sensor is 3d positions of 20 body joints for each person defined in the global coordinate. We define our skeleton representation for the first person as $S^{(1)} = (V^{(1)}, E^{(1)})$, where $V^{(1)} = \{v_1^{(1)}, v_2^{(1)}, \ldots, v_N^{(1)}\}, N = 20$ denotes the set of joints and $E^{(1)} = \{e_1^{(1)}, e_2^{(1)}, \ldots, e_M^{(1)}\}, M = 19$ denotes the set of body parts (See Fig.1). Let $e_{m1}^{(1)}, e_{m2}^{(1)} \in \mathcal{R}^3$ represents the starting and ending points of the body part $e_m^{(1)}$. Similarly we have $S^{(2)} = (V^{(2)}, E^{(2)})$ for the second person.

First, we define the person specific local coordinate at the first person's shoulder center computed as $(v_5 + v_6)/2$ such that the x coordinate is aligned with the person's body orientation. Then we update both $S^{(1)} = (V^{(1)}, E^{(1)})$ and $S^{(2)} = (V^{(2)}, E^{(2)})$ using this new coordinates. This process makes our representation invariant to the global translation and orientation.

Next we consider a set of body parts, $E = \{E^{(1)}, E^{(2)}\}$, and for each pair of body parts e_m and e_n, in E, we describe their relative geometry by the translation and rotation. The translation is computed as $T_{m,n} = e_{m1} - e_{n1}$. For the rotation, we compute the rotation axis r and the rotation angle θ between two vectors $e_{n2} - e_{n1}$ and $e_{m2} - e_{m1}$. From them we compute a quaternion $q_{m,n} \in \mathcal{R}^4$ by $q_1 = r_1 sin(\frac{\theta}{2}), q_2 = r_2 sin(\frac{\theta}{2}), q_3 = r_3 sin(\frac{\theta}{2}), q_4 = cin(\frac{\theta}{2})$.

Thus, each pair of parts can be represented as 7 dimensional vector. Since there are $\binom{38}{2}$ such pairs, we have $\binom{38}{2} \times 7 = 4921$ dimensional vector. We do the same step by using the second person's local coordinate and concatenate two vectors to obtain the final feature vector representing two people configurations at a current frame.

We compute our skeleton feature across 41 frames which includes the action center frame, 20 frames before the action center frame and 20 frames after the action center frame. Finally we concatenate all the feature vectors into a final vector.

5. Action Classification

We propose two different methods for classifying interactive actions from the skeleton features.

5.1. Generative Modeling

Given an input video containing the action, generate a semantic desription of the action as it occurs in the video. To describe an action would mean to generate a dictionary based representation for the action. A suitable middle ground between a complete linguistic action definition and a
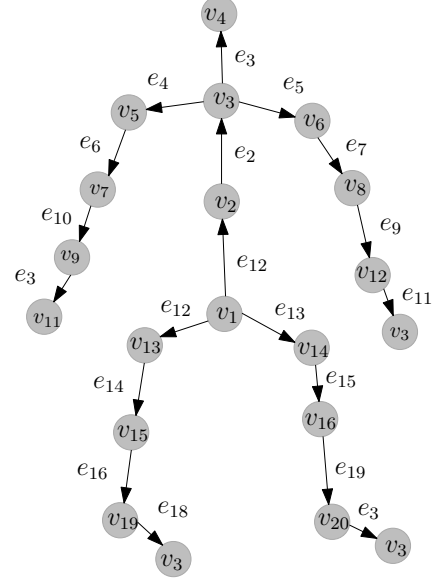

Figure 1: An illustration of the skeleton.

machine-derivable description, we borrow the action vocabulary decribed in [2]. Accordingly, a single person action in general can be described using an **operational triplet** as follows:

$$< Agent - Motion - Target > \qquad (1)$$

where $Agent$ is the person or the body part of the person which initiates the action; $Motion$ implies the movement of the $Agent$; $Target$ can be taken to be an object or another person or his body part that is involved in the interaction. Using this terminology some simple action representations can be generated as shown below:

- **Handshake**: <Arm1 - Outstretched - Arm2>

- **Kicking**: <Leg1 - Outstretched - Leg2>

It should be noted that for some actions that *Agent* and *Target* need not necessarily be a single person or body part but can be a collection of body parts that act on each other in a causal way. To see an example of this, consider the figure **??**

Thus, we can see that actions that have similar semantic descriptions in terms of the operational triplet can be differentiated by means of proper temporal constraints. To look at how spatial constraints can decide the type of interaction, we incorporate that knowledge in the decision tree structure shown in figure **??**.
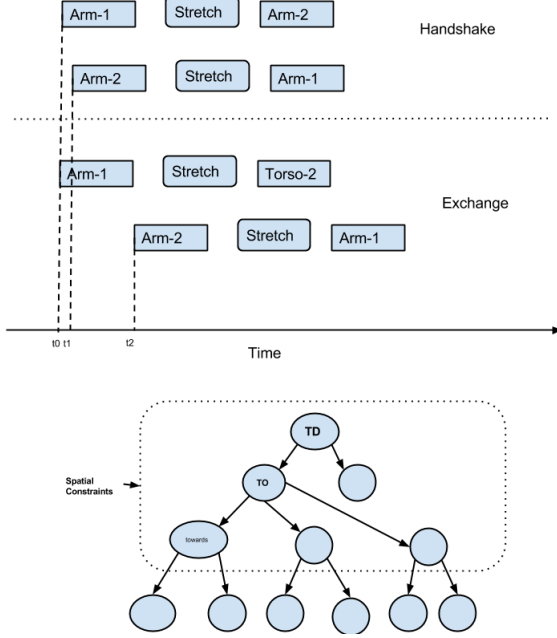
5.2. Artificial Feed-forward Neural Network

We use artificial feed-forward neural network with softmax activation function as its final layer to estimate conditional probability, $P(y_t|x_t, t = t_{ac})$ at time instance t, where y_t is an action class label and x_t is the skeleton features computed from $t - 20$ to $t + 20$ frames and t_{ac} is the time instance of the action center frame. We manually localize the action center frame of the training sequences and train the model with 10 hidden layers.

In testing time, we apply the trained neural network in a time sliding window manner and obtain the conditional probability $P(y_t|x_t, t = t_{ac})$. We also apply our action center detector at each frame to obtain $P(t = t_{ac}|x_t)$. We then compute $P(y_t, t = t_{ac}|x_t) = P(y_t|x_t, t = t_{ac})P(t = t_{ac}|x_t)$ at each frame. Finally, we compute $P(y)$ for each sequence by $\sum_{t=t_s}^{t_e} P(y_t, t = t_{ac}|x_t)$ where t_s and t_e are the time instance where there are sufficient number of frames to apply the neural network. The final prediction is done by $\operatorname{argmax}_y P(y)$.

6. Experiments

6.1. Data Collection

We create a new dataset named Maryland Interactive Action Dataset (MIAD) which consists of 10 interactive action classes captured by Kinect. The action classes we consider are summarized in Fig.2 with their representative images. For each pair of people, we collect 2 sequences per an action class by switching their locations. Since we collect data from 6 pairs of people, there are 12 sequences per action class. We use the first X sequences as training data and the remaining sequences as testing data.

Our new data, MIAD differs from the existing two people interactive dataset such as K3HI [3] and SBU Kinect Interaction Dataset [4]. First, our skeleton is represented by 20 joints whereas K3HI and SBU have only 15 joints. Second, the action classes we consider are more 'fine-grained'. In K3HI and SBU, there is only one action class for 'exchanging an object' class while MIAD contains 5 unique action classes for it, differentiated by the object being exchanged. Also K3HI and SBU have only one action class for the greeting action, namely, shaking hands, while MIAD includes 5 different action classes. These fine-grained actions are generally more challenging to discriminate as actions are more similar each other, necessitating more detailed representation of body movements.

6.2. Action Localization Results

Show the examples of the starting frame, center frame and the ending frame.

6.3. Action Classification Results

Show confusion matrix. Give some discussion on difficult action classes.

7. Conclusion

References

[1] Raviteja Vemulapalli, Felipe Arrate, Rama Chellappa, *Human Action Recognition by Representing 3D Skeletons as Points in a Lie Group*. CVPR, 2014. 1

[2] Semantic-level Understanding of Human Actions and Interactions using Event Hierarchy, Sangho Park, J.K. Aggarwal, CVPR'04 2

[3] Sangho Park, J.K. Aggarwal, *K3HI: Kinect-based 3D Human Interaction Dataset* 3

[4] Kiwon Yun, Jean Honorio, Debaleena Chattopadhyay, Tamara L. Berg, and Dimitris Samaras, *Two-person Interaction Detection Using Body-Pose Features and Multiple Instance Learning*. CVPRW, 2012.
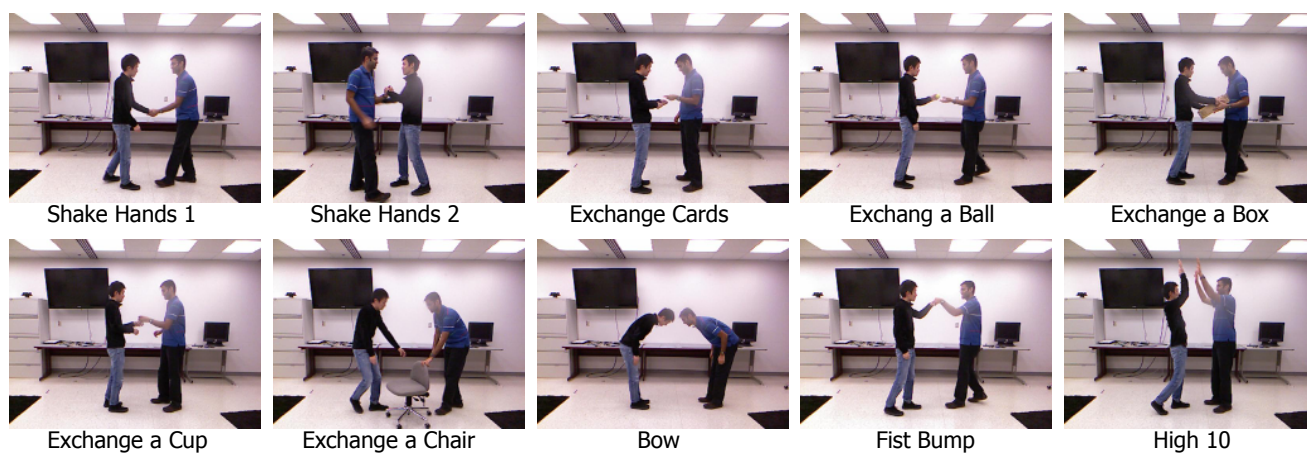
Figure 2: Action classes in newly collected MIAD dataset.