Two-person Interaction Detection Using Body-Pose Features and Multiple Instance Learning

Kiwon Yun[1], Jean Honorio[1], Debaleena Chattopadhyay[2], Tamara L. Berg[1], Dimitris Samaras[1]
[1]Stony Brook University, Stony Brook, NY 11794, USA
[2]Indiana University, School of Informatics at IUPUI, IN 46202, USA
{kyun, jhonorio, tlberg, samaras}@cs.stonybrook.edu, debchatt@iupui.edu

Abstract

*Human activity recognition has potential to impact a wide range of applications from surveillance to human computer interfaces to content based video retrieval. Recently, the rapid development of inexpensive depth sensors (*e.g. *Microsoft Kinect) provides adequate accuracy for real-time full-body human tracking for activity recognition applications. In this paper, we create a complex human activity dataset depicting two person interactions, including synchronized video, depth and motion capture data. Moreover, we use our dataset to evaluate various features typically used for indexing and retrieval of motion capture data, in the context of real-time detection of interaction activities via Support Vector Machines (SVMs). Experimentally, we find that the geometric relational features based on distance between all pairs of joints outperforms other feature choices. For whole sequence classification, we also explore techniques related to Multiple Instance Learning (MIL) in which the sequence is represented by a bag of body-pose features. We find that the MIL based classifier outperforms SVMs when the sequences extend temporally around the interaction of interest.*

1. Introduction

Human activity recognition is an important field for applications such as surveillance, human-computer interface, content-based video retrieval, etc. [1, 26]. Early attempts at human action recognition used the tracks of a person's body parts as input features [7, 35]. However, most recent research [14, 6, 23, 30] moves from the high-level representation of the human body (*e.g.* skeleton) to the collection of low-level features (*e.g.* local features) since full-body tracking from videos is still a challenging problem. Recently, the rapid development of depth sensors (*e.g.* Microsoft Kinect) provides adequate accuracy of real-time full-body tracking with low cost [31]. This enables us to once again explore the feasibility of skeleton based features for activity recognition.

Past research proposed algorithms to classify short videos of simple periodic actions performed by a single person (*e.g.* 'walking' and 'waiving') [23, 4]. In real-world applications, actions and activities are seldom periodic and are often performed by multiple persons (*e.g.* 'pushing and 'hand shaking) [28]. Recognition of complex non-periodic activities, especially interactions between multiple persons, will be necessary for a number of applications (*e.g.* automatic detection of violent activities in smart surveillance systems). In contrast to simple periodic actions, the study of causal relationships between two people, where one person moves, and the other reacts, could help extend our understanding of human motion.

In this work, we recognize interactions performed by two people using RGBD (*i.e.* color plus depth) sensor. Recent work [22, 16, 2] has suggested that human activity recognition accuracy can be improved when using both color images and depth maps. On the other hand, it is known that a human joint sequence is an effective representation for structured motion [8]. Hence we only utilize a sequence of tracked human joints inferred from RGBD images as a feature. It is interesting to evaluate body-pose features motivated from motion capture data [20, 12, 21] using tracked skeletons from a single depth sensor. Since full-body tracking of humans from a single depth sensor contains incorrect tracking and noise, this problem is somewhat different from scenarios with clean motion capture data.

In this paper, we create a new dataset for two-person interactions using an inexpensive RGBD sensor (Microsoft Kinect). We collect eight interactions: *approaching, departing, pushing, kicking, punching, exchanging objects, hugging, and shaking hands* from seven participants and 21 pairs of two-actor sets. In our dataset, color-depth video and motion capture data have been synchronized and annotated with action label for each frame.

Moreover, we evaluate several geometric relational body-pose features including *joint features, plane features* *and velocity features* using our dataset for real-time interaction detection. Experimentally, we find *joint features* to outperform others for this dataset, whereas *velocity features* are sensitive to noise, commonly observed in tracked skeleton data.

Real time human activity detection has multiple uses from human computer interaction systems, to surveillance, to gaming. However, non-periodic actions no always have a clearly defined beginning and ending frame. Since recorded sequences are manually segmented and labeled in training data, a segmented sequence might contain irrelevant actions or sub-actions. To overcome this problem, we use the idea of Multiple Instance Learning (MIL) to tackle irrelevant actions in whole sequence classification. We find that classifiers based on Multiple Instance Learning, have much higher classification accuracy when the training sequences contain irrelevant actions than Support Machine Machine (SVM) classifiers.

This paper is organized as follows: Related work is reviewed in Section 2. Section 3 provides a detailed description of our interaction dataset. In Section 4, we define the geometric relational body-pose features for real-time interaction detection. We describe how MILBoost scheme [34] improves the performance on whole sequence classification in Section 5. Section 6 shows the experimental results and Section 7 concludes the paper.

2. Related Work

Interaction dataset: Very few person-to-person interaction dataset are publicly available. There are certain interaction dataset in video for surveillance environment [29, 28], TV shows [25], and YouTube or Google videos [13]. However, these datasets only contain videos since they focus on robust approaches in natural and unconstrained videos. There also exist motion capture datasets containing human interactions such as The CMU Graphics Lab Motion Capture Database (http://mocap.cs.cmu.edu/) and Human Motion Database (HMD) [9]. However, both datasets have only captured one couple (=two-actor set) so that they are not well suited for evaluating human interaction recognition performance. There are some datasets for pose estimation [33, 17], containing some human-human interaction sequences with videos and synchronized motion capture data. However, since the purpose of these datasets is pose estimation or shape reconstruction, they are not be directly used for activity recognition.

Kinect activity dataset: Recently, several activity recognition datasets have been released. These datasets are focused on simple activities or gestures [19, 16, 2], or daily activities [22, 32] performed by a single actor such as drinking water, cooking, entering the room, etc.

Acitivity recognition with Kinect: We briefly mention approaches to the single or daily activity recognition problem on Kinect dataset. Li *et al*. [16] use an expandable graphical model, called an action graph, to explicitly model the temporal dynamics of the actions, and a bag of 3D points extracted from the depth map to model the postures. Ni *et al*. [22] proposed multi-modality fusion schemes combining color and depth information for daily activity recognition. Both papers limit input to color and depth maps. Only Masood *et al*. [19] and Sung *et al*. [32] use joint sequences from depth sensors as a feature. In [19], only skeleton joints are used as a feature for real-time single activity recognition and actions are detected by logistic regression. However, action categories are chosen from gestures for playing video games, and can easily be discriminated from each other using a single pose. In [32], both color and depth, and skeleton joints are used as features and daily activities are classified by a hierarchical maximum entropy Markov model (MEMM). However, the action classes do not have significant motion and skeleton features they use are highly dependent on given action classes.

Multiple Instance Learning: Multiple Instance Learning (MIL) is a variant of supervised learning. In MIL, samples are organized into “bag”, instead of using positive or negative singletons, and each bag may contain many instances [18]. Recent works [11, 3, 10] show MIL provides better human action recognition and detection accuracy. MILBoost proposed by [34] use MIL in a boosting framework, and it has been successfully applied for human detection [5] and video classification [15].

3. A Two-person Interaction Dataset

We collect two person interactions using the Microsoft Kinect sensor. We choose eight types of two-person interactions, motivated by the activity classes from [29, 28, 24], including: *approaching, departing, pushing, kicking, punching, exchanging objects, hugging, and shaking hands*. Note that all of these action categories have interactions between actors that differ from the categories performed by a single actor independently. These action categories are challenging because they are not only non-periodic actions, but also have very similar body movements. For instance, ‘exchanging object’ and ‘shaking hands’ contain common body movements, where both actors extend and then withdraw arms. Similarly, ‘pushing’ might be confused with ‘punching’.

All videos are recorded in the same laboratory environment. Seven participants performed activities and the dataset is composed 21 sets, where each set contains videos of a pair of different persons performing all eight interactions. Note that in most interactions, one person is acting and the other person is reacting. Each set contains one or two sequences per action category. The entire dataset has a total of 300 interactions approximately.

Both color image and depth map are 640 × 480 pixels.

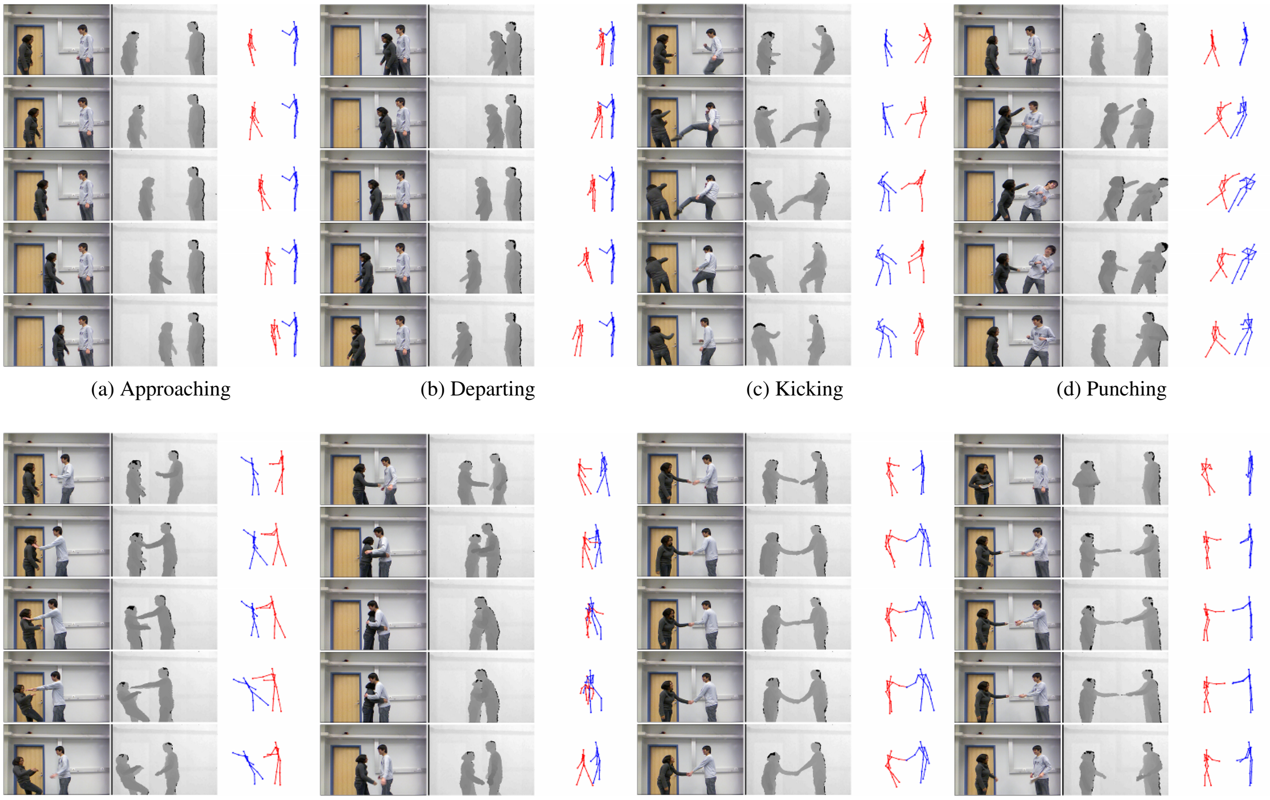


Figure 1: Visualization of our interaction dataset. Each row per interaction contains a color image, a depth map, and extracted skeletons at the first, 25%, 50%, 75%, and the last frame of the entire sequence for each interaction: *approaching, departing, kicking, punching, pushing, hugging, shaking hands, and exchanging*. A red skeleton indicates the person who is acting, and a blue skeleton indicates the person who is reacting.

The dataset apart from an image and a depth map also contains 3-dimensional coordinates of 15 joints from each person at each frame. The articulated skeletons for each person are automatically extracted by OpenNI with NITE middleware provided by PrimeSense [27]. The frame rate is 15 frames per second (FPS). The dataset is composed of manually segmented videos for each interaction, but each video roughly starts from a standing pose before acting and ends with a standing pose after acting. Our dataset also contains ground truth labels with each segmented video labeled as one action category. Ground truth label also contains identification of "active" actor (*e.g.* the person who is punching), and "inactive" actor (*e.g.* the person being punched). Figure 1 shows example snapshot images of our dataset.

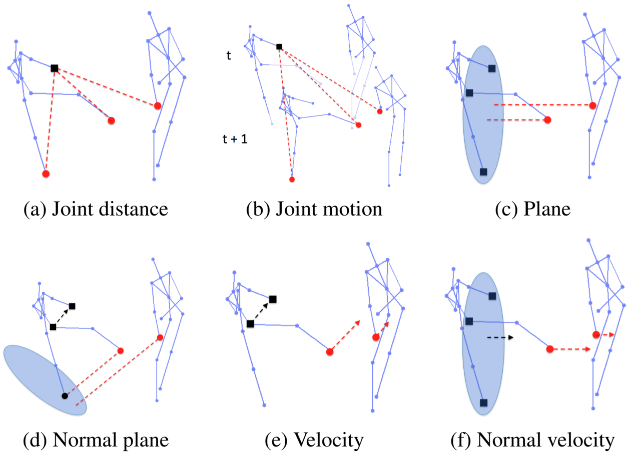
Although the skeleton extraction from depth maps provides a rather accurate articulated human body, it contains noisy and incorrect tracking. Especially, since the full-body tracking by NITE middleware is less stable on fast and complex motions, and occlusions [27], there often exist tracking failures in our dataset. For example, the position of an arm is stuck in Figure 1e and Figure 1a. The overall tracking is sometimes bad when a large amount of body parts of two persons overlap (*e.g.* Figure 1f). More examples can be found in the supplementary material.

4. Evaluation of Body-Pose Features for Real-time Interaction Detection

In this section, we utilize several body-pose features used for indexing and retrieval of motion capture data, and evaluate them using our dataset for real-time detection of interaction activities. Here, real-time refers to recognition from a very small window of 0.1-0.2 seconds (2-3 frames). Interaction detection is done by Support Vector Machine (SVM) classifiers. In what follows, we describe the features under our evaluation.

4.1. Features

One of the biggest challenges of using skeleton joints as a feature is that semantically similar motions may not necessarily be numerically similar [21]. To overcome this, [36] uses relational body-pose features introduced in [21]



(a) Joint distance (b) Joint motion (c) Plane

(d) Normal plane (e) Velocity (f) Normal velocity

Figure 2: Body-pose features. Black rectangle indicates a reference joint or vector, red circle indicates a target joint, and blue circle indicates a reference plane. Red line is computed by the definition of features and only two or three samples are shown here.

describing geometric relations between specific joints in a single pose or a short sequence of poses. They use relational pose features to recognize daily-life activities performed by a single actor in the random forest framework. We design a number of related features for two-person interaction recognition and evaluate them on our dataset, with a small window size (2-3 frames).

Let $p^x_{i,t} \in \Re^3$ and $v^x_{i,t} \in \Re^3$ be the 3D location and velocity of joint i of person x at time t. Let T be all frames within the size of window, W. Each window is labeled as the action being executed in the middle. A feature of each window is a single vector as a concatenation of all computed features $F(\cdot;t)$, where $t \in T$.

Joint distance: The *joint distance* feature F^{jd} (see Figure 2a) is defined as the Euclidean distance between all pairs of joints of two persons at time t. It captures the distance between two joints in a single pose. It is defined as:

$$F^{jd}(i,j;t) = \|p^x_{i,t} - p^y_{j,t}\|, \tag{1}$$

where i and j are any joints of two persons, $t \in T$, and this is measured for one person ($x = y$) or between two persons ($x \neq y$).

Joint motion: The *joint motion* feature F^{jm} (see Figure 2b) is defined as the Euclidean distance between all pairs of joints of two persons between at time t_1 and at time t_2. It captures dynamic motions of two persons at time t_1 and t_2. It is defined as:

$$F^{jm}(i,j;t_1,t_2) = \|p^x_{i,t_1} - p^y_{j,t_2}\|, \tag{2}$$

where i and j are any joints of two persons, $t_1, t_2 \in T$, $t_1 \neq t_2$, and this is measured for one person ($x = y$) or between two persons ($x \neq y$).

Plane: The *plane* feature F^{pl} (see Figure 2c) captures the geometric relationship between a plane and a joint. For example, one may express how far the right foot lie in front of the plane spanned by the left knee, the left hip and the torso for a fixed pose. It is defined as:

$$F^{pl}(i,j,k,l;t) = dist(p^x_{i,t}, \langle p^y_{j,t}, p^y_{k,t}, p^y_{l,t}\rangle), \tag{3}$$

where $\langle p^y_{j,t}, p^y_{k,t}, p^y_{l,t}\rangle$ indicates the plane spanned by p^y_j, p^y_k, p^y_l, and dist$(p^x_i, \langle\cdot\rangle)$ is the closest distance from point p^x_j to the plane $\langle\cdot\rangle$. $t \in T$, and this is measured for one person ($x = y$) or between two persons ($x \neq y$).

Normal plane: The *normal plane* feature F^{np} (see Figure 2d) captures something the *plane* feature cannot express. For example, using the plane that is normal to the vector from the joint 'neck' to the joint 'torso', one can easily check how far a hand raised above neck height. It is defined as:

$$F^{np}(i,j,k,l;t) = dist(p^x_{i,t}, \langle p^y_{j,t}, p^y_{k,t}, p^y_{l,t}\rangle_n), \tag{4}$$

where $\langle p^y_{j,t}, p^y_{k,t}, p^y_{l,t}\rangle_n$ indicates the plane with normal vector $p^y_j - p^y_k$ passing through p^y_l. As in the plane feature, $t \in T$, and this is measured for one person ($x = y$) or between two persons ($x \neq y$).

Velocity: The *velocity* feature F^{ve} (see Figure 2e) captures the velocity of one joint along the direction between two other joints at time t. It is defined as:

$$F^{ve}(i,j,k;t) = \frac{v^x_{i,t} \cdot (p^y_{j,t} - p^y_{k,t})}{\|p^y_{j,t} - p^y_{k,t}\|}, \tag{5}$$

where $t \in T$, and this is measured for one person ($x = y$) or between two persons ($x \neq y$).

Normal velocity: The *normal velocity* feature F^{ve} (see Figure 2f) captures the velocity of one joint in the direction of the normal vector of the plane spanned by three other joints at time t. It is defined as:

$$F^{nv}(i,j,k,l;t) = v^x_{i,t} \cdot \hat{n}\langle p^y_{j,t}, p^y_{k,t}, p^y_{l,t}\rangle, \tag{6}$$

where $\hat{n}\langle\cdot\rangle$ is the unit normal vector of the plane $\langle\cdot\rangle$, $t \in T$, and this is measured for one person ($x = y$) or between two persons ($x \neq y$).

5. Interaction Recognition on Whole Action Sequences via Multiple Instance Learning

In the previous section we considered classification of short sequences (2-3 frames) directly centered around the peak of the interaction of interest. In this section, we explore what happens for longer time frames. As we explain in Section 3, each video in training data is manually segmented from the start frame, when 'active' actor starts to move from a standing pose, to the end frame, when both 'active' and 'inactive' actor go back to a standing pose. For instance, a segmented video for the 'hugging' action starts from when both actors start to approach each other. It ends when they stand apart each other after hugging. Thus, the hugging video contains earlier and later frames which can be irrelevant of the 'hugging' action, and might be more similar to approaching and departing. Standard classifiers learned on sequences like these will have low accuracy.

We use Multiple Instance Learning(MIL) in a boosting framework to handle irrelevant frames in the training data. Multiple Instance Boosting (MILBoost) proposed by Viola *et al*. [34] was successfully applied to face detection [34], human detection [5] and video classification [15]. MILBoost combines AnyBoost with MIL. In boosting framework, each instance is classified by a linear combination of weak classifiers. In MILBoost, each instance is not individually labeled. Instead, training instances are organized into bags of instances. A positive bag has at least one positive instance in the bag, while all instances in a negative bag are negative instance. In selecting the weak learner, MILBoost will pay more attention to instances that have higher weight. Thus, the algorithm assigns a higher positive weight on a subset of instances, and these instances dominate subsequence learning.

We follow the MILBoost formulation of [34] with Noisy OR model, and consider the body-pose feature at each frame as an instance. Figure 3 shows how the algorithm works to recognize 'kicking' action. MILBoost assigns higher weights on 'actual' kicking action among other instances in the bag so that it reduces the effect of irrelevant actions in the bag.

6. Experiments

In this section, we first evaluate body-pose features for real-time interaction detection. Second, we classify segmented sequences into action labels using MILBoost and compare the result with using SVMs.

6.1. Experiments for Real-time interaction detection

The features defined in Section 4.1 can be divided into three groups: two *joint features*, two *plane features*, and two *velocity features*. We evaluate which group of features are the most appropriate for real-time two-person interaction detection. 30 joints (*i.e.* 15 joints of each person) are used for *joint features*. Thus the dimension of the *joint distance* feature is W × 435 for each frame. The *joint motion* feature has a higher dimension (*i.e.* $30 \times \binom{W}{2}$). Both *plane features* and *velocity features* are much higher dimension vectors. For this reason, we choose ten markers (*i.e.* 'torso',

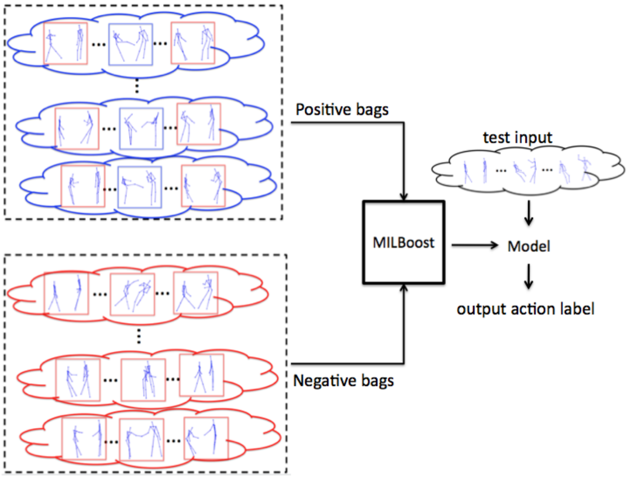


Figure 3: The overview of the 'kicking' MILBoost classifier. Blue rectangles indicate true positive instances and red rectangles indicate true negative instances. If a positive bag has at least one instance is positive (*i.e.* actual kicking action), while a negative bag does not have any of kicking action. The MILBoost classifier outputs an action label given a test sequence.

'head', 'elbows', 'hands', 'knees' and 'feet') for a target joint (*i.e.* i in Equation 1 and 2), while selecting only six important markers (*i.e.* 'torso', 'head', 'hands' and 'feet') for reference planes or vectors (*i.e.* j, k, l in Equation 3, 4, 5, and 6). By doing so, we create a lower dimension feature without losing meaningful information. However, both *plane features* and *velocity features* have very high dimension (*i.e.* 800 × W for *plane* and *velocity*, even higher for *normal velocity*).

All body-pose features are computed within W=3 (0.2 seconds). To classify eight action categories, we train SVMs in a one-vs-all fashion, and evaluation is done by 5-fold cross validation, i.e. 4 folds are used for training, and 1 for testing. The dataset is composed by 21 sets of two actors. We randomly split the dataset into 5 folds of 4-5 two-actor sets each. The partitioning of the datasets into folds is performed so that each two-actor set is guaranteed to appear only in training or only in testing. Table 1 shows the real-time activity detection accuracy of eight complex human-human interactions from our dataset. The results are averaged over the 5 permutations and the parameter selection of SVMs is done by nested cross validation with cost C $\in$ {0.01, 0.1, 1, 10, 100}. We also evaluated with a non-linear kernel for the SVM, but it yielded very little, if any, improvement for most of features since our body-pose features are high dimensional. The result shows *joint features* result in higher detection accuracy than *plane features* and *velocity features*. We conclude that the geometric relational body-pose feature based on euclidean distance between joints captures the temporal and dynamic informa-

Features	Average accuracy
Raw position	0.497 ± 0.0480
Joint distance	**0.793** ± 0.0276
Joint motion	**0.802** ± 0.0390
Plane	0.612 ± 0.0282
Normal plane	0.723 ± 0.0333
Velocity	0.442 ± 0.0393
Normal Velocity	0.349 ± 0.0193
Joint features (Figure 2a & 2b)	**0.803** ± 0.0399
Plane features (Figure 2c & 2d)	0.738 ± 0.0192
Velocity features (Figure 2e & 2f)	0.484 ± 0.0387
Joint features + Plane features	0.790 ± 0.0349
Joint features + Velocity features	0.802 ± 0.0357
Velocity features + Plane features	0.744 ± 0.0201
All features	0.790 ± 0.0331

Table 1: Detection performance (± standard deviation) with various combinations of body-pose features. *Joint features* (*i.e. joint distance* and *joint motion*) are the strongest feature than others.

tion of body movement for complex human activities in the real-time action detection scenario. Also, it is more stable with noisy full-body tracking than *velocity features*. Yao *et al*. [36] pointed out *velocity features* have the best accuracy for single activity recognition using clean motion capture data with reduced skeleton (=13 joints). However, they also claimed that *velocity features* are not robust to noise by using synthesized noisy skeleton. As we have seen in Section 3, our dataset contains a significant amount of incorrect tracking and noise, which might explain the lower accuracy of *velocity features*. Note that the raw position feature uses the position of all 30 joints at a frame. The low accuracy using the raw position feature means that the actions in the dataset are difficult to classify.

Figure 4 shows confusion matrices for different body-pose features. Over all eight action categories, *joint features* have the best accuracy among three feature groups. 'Hugging' is the most confused action in all cases; it is mostly confused with 'pushing'. Note that 'hugging' tends to have more tracking and noise problems, since two skeletons overlap. As we have pointed out in Section 5, sequences in training data are manually segmented and a sequence may contain irrelevant actions. For this reason, in many cases, the beginning part of 'hugging' is classified as 'approaching' and the last part of 'hugging' is classified as 'departing'. Moreover, when two persons get close with their stretched arms, the motion is very similar as 'pushing' action. One way to model this would be to divide the initial part of the sequence into a sub-action type such as "stretch arm", which is out of the scope of the paper. We consider this the problem of irrelevant actions in training data, meaning that the part of 'approaching', 'departing', and 'stretch arm' in the 'hugging' action is not actual 'hugging' interaction, instead they are irrelevant sub-actions. We leave exploration of sub-actions for future work. For similar reasons, there also exists some confusion between 'shaking hands' and 'exchanging', and between 'pushing' and 'punching'. In real-time, their actions are very similar and it leads to a low classification accuracy. Figure 5 shows examples of real-time detection results.

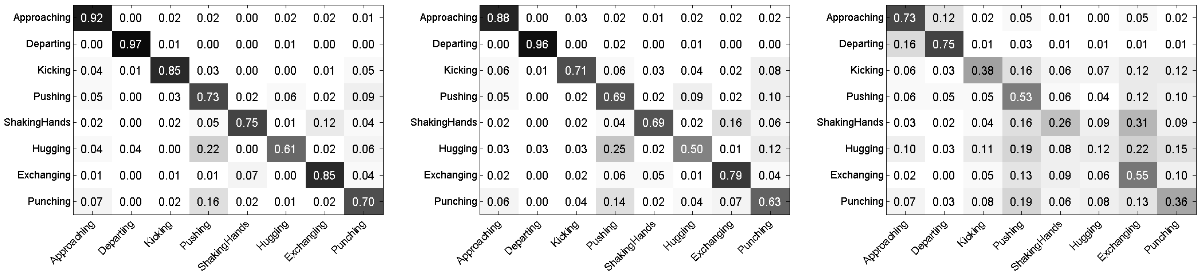
6.2. Experiments on Whole Action Sequence Classification

In this section, we compare the classification performance between the MILBoost classifier and SVM classifier with segmented videos. Note that this experiment is not real-time detection. We compute *joint distance* feature for each frame and the size of window is the maximum frame size of a sequence. For MILBoost classifier, the size of the bag is the same as the window size for SVM classifiers. In SVMs each frame is treated equally during learning, while in MILBoost frames are re-weighted so that key frames for the interaction receive larger weights. For example, a 'punching' classifier gives higher weights the feature of the frame, where the arm of 'active' actor is stretched and almost hit the head of 'inactive' actor. Since the lower weights will be assigned to the frames having irrelevant actions, the classification performance increases.

To evaluate this, we first use the sequences with our ground truth labels in the training set, called *Set 1*. To amplify the effect of irrelevant actions in the training set, we create a different training set, called *Set 2*, with more irrelevant actions. Specifically, we segment the original recorded sequence by starting from five frame earlier than the original start frame and ending five frame later than the original final frame. *Set 2* contains more irrelevant actions since participants randomly moved between action categories when we collected data. We learn SVMs and MILBoost classifiers in both *Set 1* and *Set 2* and test for whole sequence classification. We use linear SVM with cost C ∈ {0.01, 0.1, 1, 10, 100} and run 100 iteration in MILBoost. The classification performance is evaluated by 5-fold cross validation. Table 2 shows the accuracy of whole sequence classification. On *Set 1*, MILBoost has slightly better performance than SVMs. However, we found accuracy dramatically dropped with SVMs from *Set 1* to *Set 2*, while MILBoost retain high classification accuracy with training data including more irrelevant actions. We conclude the MILBoost classifier outperforms SVMs if there exist irrelevant actions in the training set.

7. Conclusion and Future Work

We have created a new dataset for two-person interaction using the Microsoft Kinect sensor including eight interactions, color-depth video and motion capture data at each frame. Using this dataset, we have evaluated body-pose



(a) Joint features

(b) Plane features

(c) Velocity features

Figure 4: Confusion matrix of different body-pose features for real-time interaction detection (W=3). Average classification rates are 80.30%, 73.80%, 48.84% respectively.

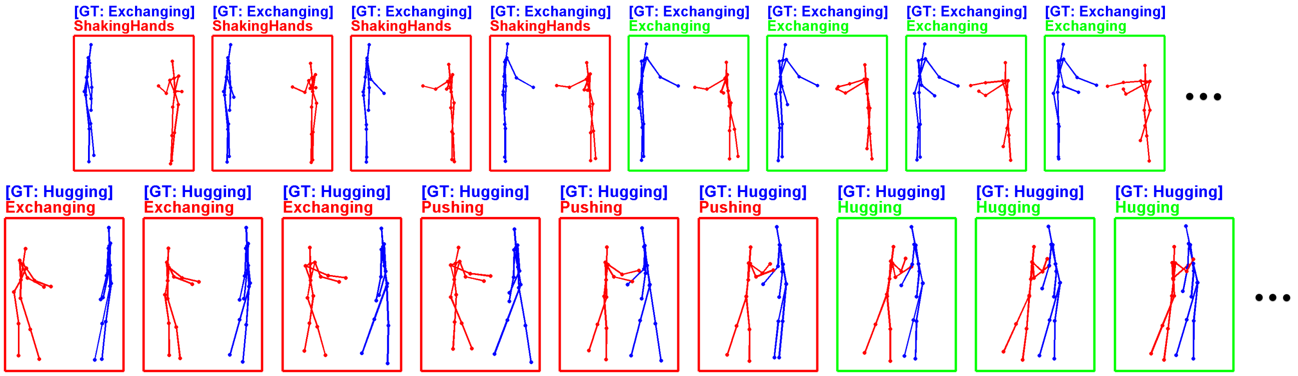


Figure 5: Examples of real-time interaction detection. Each row shows the detected activity. A green box is true detection and a red box is false detection. Each box is selected every 2-5 frames in a sequence. Ground truth label is shown in a square bracket. Top: the first few frames are incorrectly classified as 'shaking hands', instead of 'exchanging'. Bottom: the first few frames are classified as either 'exchanging' or 'pushing', not as 'hugging'. All these false detection are caused by irrelevant actions in training data. More results can be found in the supplementary material

Classifier	*Set 1*	*Set 2*	Performance decrease
Linear SVMs	0.876	0.687	-0.189
MILBoost	**0.911**	**0.873**	-0.038

Table 2: The performance on whole sequence classification. With original label, MILBoost has better classification result than SVMs. With more irrelevant actions in the training data, the performance of SVMs is dramatically dropped, while MILBoost retain high accuracy.

features motivated from 3D skeleton features for indexing and retrieval of motion capture data. Geometric relational features based on distance between all pairs of joints (*i.e. joint features*) outperformed other features for real-time interaction detection on noisy 3D skeleton data. Moreover, we have shown that the MILBoost classifier outperforms SVMs if there exist irrelevant actions in the training data.

In the future, we plan to extend our interaction dataset to include additional interaction categories. One limitation of our current dataset is that all videos are captured from a specific viewpoint. In the future, we plan to extend our dataset with multiple viewpoints. Moreover, we will explore better human interaction representations on our dataset. As in [22, 32, 37], combined features with color and depth (*e.g.* video + depth + skeleton) can also be evaluated. The segmentation of sub-actions is also one interesting possible line of work. In addition, we would like to investigate how body parts of two actors relate to each other temporally during complex body movements. We expect it should be possible to find causal relationships between two actors during interactions, where one actor moves and the other reacts.

Acknowledgements This work was supported in part by NSF Awards IIS-1111047 and IIS-1054133.

References

[1] J. Aggarwal and M. Ryoo. Human activity analysis: A review. In *ACM Computing Surveys*, 2011. 1

[2] M. Alcoverro, A. Lopez-Mendez, M. Pardas, and J. Casas. Connected operators on 3d data for human body analysis. In *CVPR Workshops*, 2011. 1, 2

[3] S. Ali and M. Shah. Human action recognition in videos using kinematic features and multiple instance learning. *IEEE TPAMI*, 32(2):288–303, Feb. 2010. 2

[4] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *In ICCV*, 2005. 1

[5] Y.-T. Chen, C.-S. Chen, Y.-P. Hung, and K.-Y. Chang. Multi-class multi-instance boosting for part-based human detection. In *ICCV Workshops*, 2009. 2, 5

[6] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. *In 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, PETS*, 2005. 1

[7] D. M. Gavrila and L. S. Davis. Towards 3-d model-based tracking and recognition of human movement: a multi-view approach. In *International Workshop on Automatic Face- and Gesture-Recognition. IEEE Computer Society*, 1995. 1

[8] J. Gu, X. Ding, S. Wang, and Y. Wu. Action and gait recognition from recovered 3-d human joints. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 2010. 1

[9] G. Guerra-Filho and A. Biswas. A human motion database: The cognitive and parametric sampling of human motion. In *FG*, 2011. 2

[10] Y. Hu, L. Cao, F. Lv, S. Yan, Y. Gong, and T. Huang. Action detection in complex scenes with spatial and temporal ambiguities. In *ICCV*, 2009. 2

[11] N. Ikizler-Cinbis and S. Sclaroff. Object, scene and actions: Combining multiple features for human action recognition. *In ECCV*, 2010. 2

[12] L. Kovar and M. Gleicher. Automated extraction and parameterization of motions in large data sets. *ACM Trans. Graph.*, 23(3):559–568, Aug. 2004. 1

[13] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: A large video database for human motion recognition. In *ICCV*, 2011. 2

[14] I. Laptev, M. Marszalek, C. Schmid, and R. Benjamin. Learning realistic human actions from movies. *In CVPR*, 2008. 1

[15] T. Leung, Y. Song, and J. Zhang. Handling label noise in video classication via multiple instance learning. In *ICCV*, 2011. 2, 5

[16] W. Li, Z. Zhang, and Z. Liu. Action recognition based on a bag of 3d points. In *CVPR Workshops*, 2010. 1, 2

[17] Y. Liu, C. Stoll, J. Gall, H.-P. Seidel, and C. Theobalt. Markerless motion capture of interacting characters using multi-view image segmentation. In *CVPR*, 2011. 2

[18] O. Maron and T. Lozano-Prez. A framework for multiple-instance learning. In *NIPS*, 1998. 2

[19] S. Masood, C. Ellis, A. Nagaraja, M. Tappen, J. LaViola, and R. Sukthankar. Measuring and reducing observational latency when recognizing actions. In *ICCV Workshops*, 2011. 2

[20] M. Müller, A. Baak, and H.-P. Seidel. Efficient and robust annotation of motion capture data. In *SCA*, pages 17–26, New York, NY, USA, 2009. ACM. 1

[21] M. Müller, T. Röder, and M. Clausen. Efficient content-based retrieval of motion capture data. *ACM Trans. Graph.*, 24(3):677–685, 2005. 1, 3

[22] B. Ni, G. Wang, and P. Moulin. Rgbd-hudaact: A color-depth video database for human daily activity recognition. In *ICCV Workshops*, 2011. 1, 2, 7

[23] J. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *IJCV*, 79(3):299–318, 2008. 1

[24] S. Park and J. Aggarwal. Recognition of two-person interactions using a hierarchical bayesian network. In *First ACM SIGMM international workshop on Video surveillance*. ACM, 2003. 2

[25] A. Patron-Perez, M. Marszalek, A. Zisserman, and I. Reid. High five: Recognising human interactions in tv shows. *In BMVC*, 2010. 2

[26] R. W. Poppe. A survey on vision-based human action recognition. In *Image and Vision Computing*, 2010. 1

[27] PRIMESENSE. http://www.primesense.com, 2010. 3

[28] M. S. Ryoo and J. K. Aggarwal. Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In *ICCV*, 2009. 1, 2

[29] M. S. Ryoo and J. K. Aggarwal. UT-Interaction Dataset, ICPR contest on Semantic Description of Human Activities (SDHA). http://cvrc.ece.utexas.edu/SDHA2010/Human_Interaction.html, 2010. 2

[30] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: a local svm approach. *In International Conference on Pattern Recognition, ICPR*, 2004. 1

[31] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, 2011. 1

[32] J. Sung, C. Ponce, B. Selman, and A. Saxena. Human activity detection from rgbd images. In *AAAI workshop on Pattern, Activity and Intent Recognition (PAIR)*, 2011. 2, 7

[33] N. van der Aa, X. Luo, G. Giezeman, R. Tan, and R. Veltkamp. Umpm benchmark: A multi-person dataset with synchronized video and motion capture data for evaluation of articulated human motion and interaction. In *ICCV Workshops*, 2011. 2

[34] P. Viola, J. Platt, and C. Zhang. Multiple instance boosting for object detection. *In NIPS*, 2006. 2, 5

[35] Y. Yacoob and M. Black. Parameterized modeling and recognition of activities. In *ICCV*, 1998. 1

[36] A. Yao, J. Gall, G. Fanelli, and L. V. Gool. Does human action recognition benefit from pose estimation? *In BMVC*, 2011. 3, 6

[37] H. Zhang and L. E. Parker. 4-dimensional local spatio-temporal features for human activity recognition. In *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, 2011. 7