EUROGRAPHICS 2017 / L. Barthe and B. Benes
(Guest Editors)

Volume 36 (2017), Number 2

Character-Object Interaction Retrieval using the Interaction Bisector Surface

X. Zhao[1], M.G. Choi[2], T. Komura[3]

[1]Xi'an Jiaotong University, China
[2]The Catholic University of Korea, Korea
[3]School of Informatics, University of Edinburgh, UK

Abstract

In this paper, we propose a novel approach for the classification and retrieval of interactions between human characters and objects. We propose to use the interaction bisector surface (IBS) between the body and the object as a feature of the interaction. We define a multi-resolution representation of the body structure, and compute a correspondence matrix hierarchy that describes which parts of the character's skeleton take part in the composition of the IBS and how much they contribute to the interaction. Key-frames of the interactions are extracted based on the evolution of the IBS and used to align the query interaction with the interaction in the database. Through the experimental results, we show that our approach outperforms existing techniques in motion classification and retrieval, which implies that the contextual information plays a significant role for scene and interaction description. Our method also shows better performance than other techniques that use features based on the spatial relations between the body parts, or the body parts and the object. Our method can be applied for character motion synthesis and robot motion planning.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

1. Introduction

Human motion recognition is a research topic that has been tackled by researchers in areas such as computer graphics, computer vision and machine learning, for applications including human computer interaction, surveillance, sports analysis, medical diagnosis and robot motion planning. Metrics to compute the similarity between the movements are developed such that the movements can be categorized into different classes for motion recognition, or interpolated within the same class to synthesize a novel motion.

Most previous metrics used for computing the similarity of movements are based on the joint positions or joint angles of the body, which are not descriptive enough for fully understanding the scene context. In scenes of daily life, humans are usually not moving in free open space, but are interacting with other humans, holding and manipulating objects or avoiding obstacles in the scene. For fully understanding the scene context, it is necessary to use a feature that considers the spatial relations between the body parts, or between the body and the surrounding objects.

The nature of close interaction lies in how the spatial relations between the body and the object dynamically change over time. For example, for an action "pick up by both arms", it is described by the fact the arms are gradually spreading out and wound around the object while the body approaches to it. In order to automatically distinguish such interactions, the feature needs to encode such dynamically changing properties, which cannot be well described by simply comparing the joint angles or joint positions.

In this paper, we propose a novel feature that describes the dynamically changing spatial relations between a human body and an object during a close interaction. This feature can be used for automatically retrieving body-object interactions in the database. We make use of the interaction bisector surface (IBS), which is composed of a set of points that are equidistant from the body and the object. The IBS has been successfully applied for recognition and synthesis of static scenes. For achieving our objective, we make use of three informative attributes of the IBS: its geometric shape, a hierarchical structure of the body parts that compose the IBS, and its evolution over time. We show that these features can greatly distinguish the type of interactions in high fidelity even without classifying or aligning the objects. In the experimental results, we also show that our method outperforms state-of-the-art metrics used in motion retrieval techniques.

The contributions of this paper can be summarized as follows:

- a novel feature that is based on the dynamically changing spatial relations between the body and the object, and a distance

function that can be used to compute similarities of body-object interactions.
- an extensive evaluation of the proposed feature and distance function by comparing it with existing schemes.

2. Related Work

In this section, we first review techniques used in human motion retrieval, then about scene hallucination, and finally about scene retrieval and synthesis.

Features and Distance Functions in Human Motion Retrieval: Content-based human motion retrieval requires a distance function for computing the similarity of the query motion and the motions in the database. In classic approaches, the Euclidean distance of the state representation vectors such as the joint angles [KPZ*04] or joint positions [KG04] are used.

As the dimensionality of such state vectors are high, and as the distance may not necessarily reflect the perceptual similarity of the movements, methods that compute the distance over the subspace of the motions encoded in self organizing maps [WXWL09, SKK04, CCW*04], PCA [FF05], local PCA [CH05] and probabilistic PCA [DGL09] are developed.

When the motion involves a lot of close contacts between the body parts, or close interactions between multiple bodies, the contextual features based on contacts or spatial relationships between body parts start to become important features for distinguishing movements. Müller et al. [MRC05] propose manually designed discrete features, such as "hands touching the feet", or "arms crossed", which describe the relation between body parts. Gao et al. [GMCL06] apply this feature for building motion graphs. These features are good in abstracting quantitatively different movements but they must be predefined by humans in advance, and they can perform poorly for types of motions that were not expected when designing the features.

To generalize better, continuous features based on the spatial relations between the joints can be useful. Such features include Euclidean distances between joint pairs [TLKS08, LLLZ16], Laplacian coordinates [HKT10, TCLK12] and local transformations [VAC14]. Recently, Vemulapalli et al. [VAC14] apply a Lie group representation for the motion of a single character, whose elements are relative transformations (SE(3)) between pairs of body parts, and show the distance function can result in state-of-the-art performance for motion retrieval. A simple approach to enhance continuous features based on spatial relations to human-object interactions is to sample points or define a local coordinate system on the object and apply the same algorithm. However, there can be ambiguities for such selections, and we show in our experimental results that such an approach performs poorly compared to our approach.

Scene Hallucination: Recently, there is a huge interest in adding character models into geometric scenes, and methods to compute human poses given the object geometry are proposed in the last few years. The common approach for achieving such an objective is to represent the pose of the character with respect to the geometric features of the object. For example, Grabner et al. [GGVG11] build a Gaussian model to represent the spatial relationship between the character and a chair, based on the distance and intersection between the polygon meshes representing the character and the chair. Kim et al. [KCGF14] predict human pose based on the trained affordance model. Jiang et al. [JKS13] use Euclidean distance, relative angle and height distance to represent the relationship between character and point cloud environment. Savva et al. [SCH*14] decomposes objects into primitives such as cuboids and compute the spatial relationship between pose and environment segments. These methods heavily rely on the features computed for the object geometry, and the poses of the character are updated by relative vectors from these feature points, mainly by inverse kinematics. The pose can be strongly affected and distorted when there is a mismatch, and thus they may not be suitable for our purpose of comparing interactions.

Spatial Relations in Scene Retrieval and Synthesis: Finally we discuss about scene retrieval and synthesis methods that make use of spatial relations. Research about scene analysis [FH10, XMZ*14] and synthesis [YYT*11, FSH11, SXZ*12, FRS*12, CLW*14, LCK*14, MSSH14] have recently attracted researchers in computer graphics, where they can be applied for procedural scene synthesis [LCK*14]. In most of these methods, the spatial relations are described by contacts, supports and relative vectors. Such relations are descriptive enough for simple relations, though they need to consider various arbitrary constraints such as collisions for avoiding artifacts especially when objects are close to one another. Due to their simplicity, they can not encode complex spatial relations.

Zhao et al. [ZWK14] propose to use the Interaction Bisector Surface (IBS) for encoding complex relations such as a chair tucked under a desk, flowers put inside a vase or a bag hung on a hook. Hu et al. [HZvK*15, HvKW*16] further extend this idea with a feature called interaction context (ICON) to describe the functionality of an object in a 3D scene. Zhao et al. [ZHG*16] propose a method to increase a variation of scenes from an exemplar scene by fitting novel objects into a scene template described by an IBS. Our idea in this paper is to compute the dynamic evolution of the IBS during a close interaction between a human body and an object, and use it as a feature for retrieving similar interactions from the database.

The purpose of our work is similar to [PKH*16], which also focuses on the interaction of character and the environment or character and an object. Pirk et al. [PKH*16] track particles on one of the interaction parts and build a spatial and temporal representation called interaction landscapes. They compute the flow of particles with respect to the receiver object and do not consider the movement of the receiver. On the contrary, our feature is based on the movement of both objects, and computes the characteristics of the entire interaction, making it a suitable feature for interaction retrieval.

3. Representation and Distance Function

In this section we introduce our multi-resolution feature that encodes the interaction between a character and an object. Here we focus on computing the distance between two static poses during an interaction. The method starts by first computing an IBS between an object and a human character (see § 3.1). Then we build a

multi-resolution structure of the human body (see § 3.2) and project this hierarchy onto the IBS (see § 3.3). Using the geometry of the IBS and the body hierarchy that composes the IBS segments, we compute the distance between different poses of a human-object (see § 3.4).

3.1. IBS between a Human Character and an Object

Here we briefly review about the Interaction Bisector Surface (IBS) [ZWK14] that describes the nature of the interaction between two objects in proximity. The IBS is a set of points that are equidistant from the two objects. It is a subset of the external medial axis, and can also be considered as a generalized Voronoi diagram. In [ZWK14], IBS is used to encode the spatial relations between adjacent objects in a scene, and used for retrieval of 3D scenes composed of multiple objects. Here we compute the IBS between the human character and an object during an interaction to use it to compute the feature of interactions.

As in [ZWK14], we compute the IBS by using the quickhull algorithm [BDH96]. Points are sampled over the surface of the object and the character (the details of the human body structure and geometry are described in the next section), and a set of ridges that are equidistant from the sample points are extracted. Among the ridges, those produced by points from the body parts and the object are used to form the IBS. It can be considered as a skeleton structure of the open space where the interactions between character and object are occurring. Examples of IBSs for different interaction frames are shown in Fig. 1.

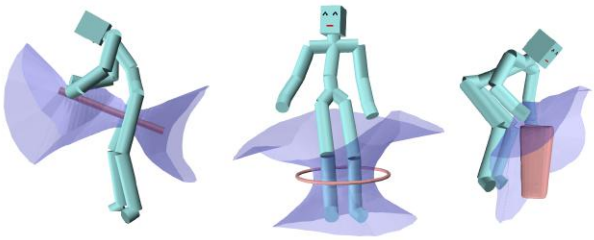

Figure 1: *IBS for different interactions. The transparent blue surface is the IBS.*

3.2. Body Hierarchy and Multi-resolution Structure

Here we describe the human body model and a multi scale structure of the body that we adopt in this paper. The same concept can be applied for different character hierarchies, such as those with fingers, though we do not use such a model due to the limitation of the capture device we use.

The skeleton structure of the body that we use in our experiments is composed of fourteen rigid body parts (as shown in Fig. 2). This skeleton is rigged to a character model whose body parts are all cylinders.

We apply a multi-resolution segmentation to the body. As we go down the multi-resoluton hierarchy, the body is divided into smaller regions. At the top level (the first level), the body is composed of

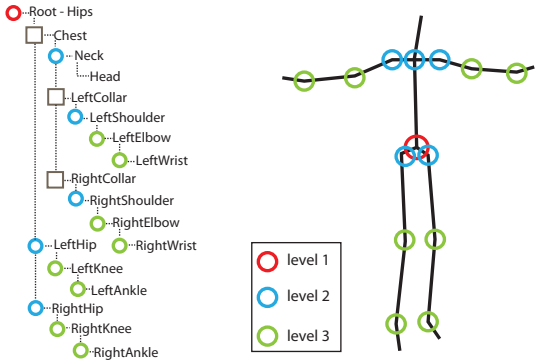

Figure 2: *BVH skeleton structure. The circles indicate the segmentation locations of the first three levels.*

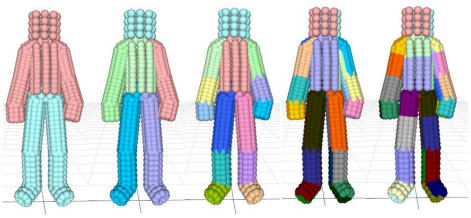

Figure 3: *The character body is segmented into 2, 6, 14, 28 and 38 regions. Each color corresponds to a unique label.*

two regions, the upper body and the lower body. In the second level, the body is divided to six regions including those of the head, torso and the four limbs. In the third level, we divide the body into individual rigid bones (i.e. the upper arm, head, thigh). We further segment the body into smaller regions along the circumference of the cylinders in the fourth level, and along the central axis of the cylinders in the fifth level (only applied to the upper arms, upper legs, and torso).

For the computation of the IBS, we uniformly sample points on the cylinder surface of the body and label each point according to the body segmentation. In Fig. 3, we show the multi-resolution structure of the body by assigning different colors to the sample points belonging to different segments.

3.3. Multi-resolution IBS Segmentation

We describe the interaction between the body and the object by examining how much each part of the body contributes to the composition of the IBS. This is done by conducting a multi-resolution segmentation of the IBS based on the multi-resolution structure of the body described above, and building a feature accordingly.

The multi-resolution segmentation of the IBS is done as follows: We visit all the ridges of IBS, and label each of them with the ID of the body segment that composes it at the bottom of the multi-resolution body structure. This can be easily found because the sample point on the body that is closest to the ridge is recorded when computing the IBS. Given the label at the bottom of the multi-

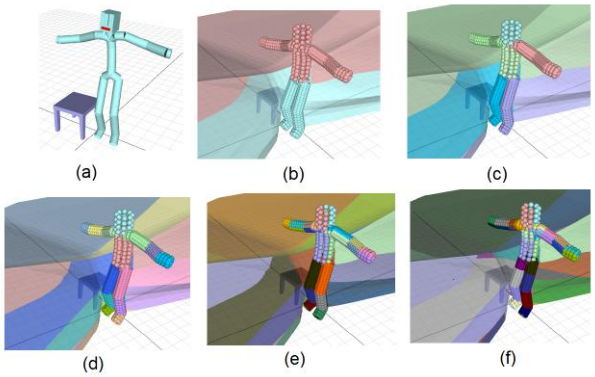

Figure 4: *A multi-resolution segmentation of the IBS based on the multi-resolution structure of the body: (a) the original scene: the character stands next to a table; (b) the grouping of the body parts at the top level of the multi-resolution structure: the upper body (red) and the lower body (green), and the label information passed to the corresponding ridges on the IBS. The red area of the IBS is composed by the upper part of the body and the green area is composed by the lower part of the body; (c)(d)(e)(f) the segmentation of the IBS by different levels of the body's multi-resolution structure.*

resolution body structure, we can know the group it belongs to at each upper level of the multi-resolution structure, and thus we can conduct the multi-resolution segmentation of the IBS. Fig. 4 shows an example IBS in a scene where the human character is standing next to a table, and its multi-resolution segmentation.

Using this multi-resolution segmentation of the IBS, we can then compute a set of feature vectors $\mathbf{V} = \{\mathbf{V}^1, \mathbf{V}^2, ..., \mathbf{V}^N\}$ where N is the number of levels of the body multi-resolution structure, which is five in our case. In the r-th level , the character body is composed of n_r parts, each of which is denoted by b_i^r $(0 \leq i < n_r)$. Let us assume that b_i^r has a corresponding region on the IBS, which is denoted as IBS_i^r (note that not all b_i^r has a corresponding IBS_i^r). The relationship feature at the r-th level, $\mathbf{V}^r$, is a n_r dimensional vector, whose entries correspond to body parts and indicate how much the body parts contribute to the interaction. The elements of each feature vector is computed as follows:

$$\mathbf{V}^r(i) = \begin{cases} \sum_{j=0}^{M} w_j & \text{if } IBS_i^r \text{ exists} \\ 0 & \text{otherwise,} \end{cases} \quad (1)$$

where M is the number of ridges in IBS_i^r, and w_j is the weight of the j_{th} ridge in IBS_i^r. w_j is computed in the same way as in [ZWK14]: $w_j = w_{\text{area}} \times w_{\text{distance}} \times w_{\text{angle}}$. Here w_{area} is the area of the ridge, w_{angle} is computed by:

$$w_{\text{angle}} = \begin{cases} 1 - \frac{\theta}{\pi/4} & \text{if } \theta < \pi/4 \\ 0 & \text{otherwise,} \end{cases} \quad (2)$$

where θ is the angle between the normal vector of the ridge and the vector from the ridge center to the corresponding point on the object, and w_{distance} is computed by:

$$w_{\text{distance}} = (1 - \frac{d}{D})^n, \quad (3)$$

where d is the distance between the ridge and the sample point, and $D = d_{\text{diag}}/2$ where d_{diag} is the length of the diagonal of the bounding box of the whole scene. We empirically set n to 20. w_{distance} is inversely proportional to the distance between the ridge and the corresponding sample point on the body, and w_{angle} is inverse proportional to the angle between the ridge normal vector and the vector from the ridge center to the corresponding sample. When $\mathbf{V}^r(i) = 0$, it means that body part b_i^r does not produce any IBS segment, and thus does not contribute to the interaction. The feature vector $\mathbf{V}$ is used to compute the distance between configurations of different interactions as described next.

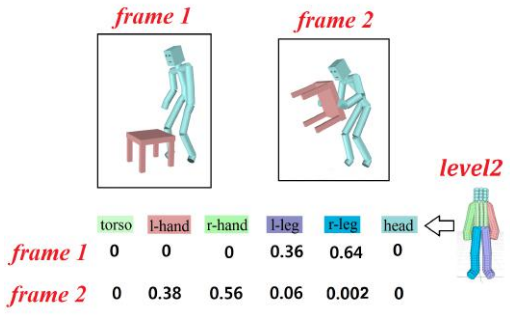

Figure 5: *Examples of proposed feature vector for two frames.*

Fig. 5 shows the feature vector at the second level of two different frames in a "pick up a table" motion.

3.4. Distance Function

We now describe how we compute the distance between two configurations represented by the multi-resolution feature vectors that we defined in the previous section.

One way to design the distance function is to quantify how much movement is needed for the body to move from one state to the other while not colliding with the object. Due to the large degrees of freedom of the human body, such a motion planning problem is extremely difficult to solve and requires a huge amount of computation. Here we provide an approximation by assuming that the object shifts along the body and the body has enough flexibility to avoid collisions when moving from one configuration to another.

Our solution is a multi-resolution distance function: multi-resolution distances are often used in computer vision for optical flow [WM95] or image matching [BSW05]. Our multi-resolution distance function is defined as follows:

$$d_{multi}(\mathbf{V}, \mathbf{V}') = \sum_{r=1}^{N} \alpha_r \cdot \frac{1}{n_r} \|\mathbf{V}^r - \mathbf{V}'^r\|, \quad (4)$$

where α_r is a weighting constant at the r-th level. It is defined as $\alpha_r = (\frac{1}{2})^r$. In all our experiments, the number of levels, N, is set to 4. So four levels of segmentation are prepared for the character body.

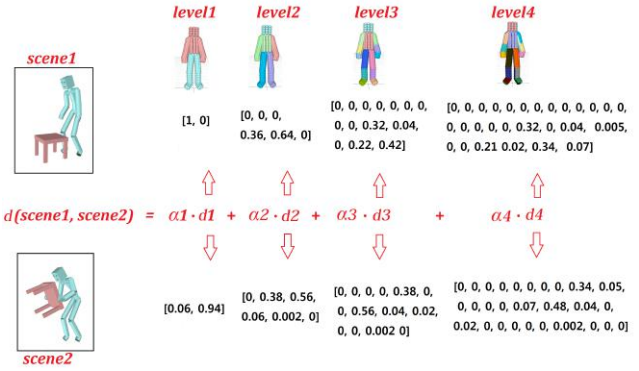

Figure 6: *The multi-resolution feature vectors and the computation of the multi-resolution distance.*

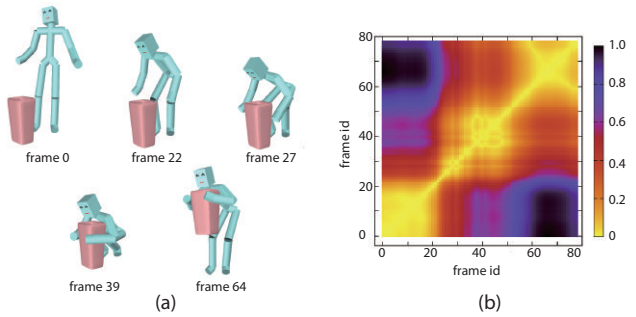

Figure 7: *(a) the "pick up the bin" motion, (b) the distance matrix of this motion. Entry $(i, j)$of the matrix is d_{multi} between frame i and frame j. The brighter color indicates lower distance.*

Fig. 6 illustrates how the distance between two configurations are computed. Even though the dissimilarity of the configurations cannot be fully described at a single level, the combination of different levels can quantify the difference in the object's configuration with respect to the human body.

To show the change of d_{multi} during a motion, we compute the distance between frames of a "pick up the bin" motion by using Eq. (4) and visualize the distance matrix in Fig. 7.

To compare two interactions, we also consider the difference between the global geometry of IBS. This is necessary as the multi-resolution feature only describes the local proximity, so that the overall difference of postures cannot be distinguished. For example, configurations shown in Fig. 8 (a) and (b) will result in similar correspondence features. To capture the global geometry of IBS, we follow Zhao et al. [ZWK14] and use the point feature histogram (PFH) [RMB*08]. The PFH is a histogram of the relative rotation between each pair of normals in the whole point cloud. It describes the local geometrical properties by generalizing the mean curvature at every point. It provides an overall pose and density invariant feature which is robust to noise.

Finally, both the local and global features of the IBS are taken into account for comparing two configurations:

$$d = w_1 \times d_{multi} + w_2 \times d_{PFH}, \tag{5}$$

where d_{PFH} is the L1 distance between the PFH features of IBS of two frames. We set $w_1 = 0.05$, $w_2 = 0.95$ in our experiment. w_1 is set small due to the large variance of the interaction feature values compared to that of PFH ($0 \leq d_{multi} \leq 0.646$, $0 \leq d_{PFH} \leq 0.0159$ in our data). For convenience, we denote our method as "MULTI+PFH" in the experiment section of the paper. In the next section, we will explain how we use this distance measure for comparing two interaction motions.

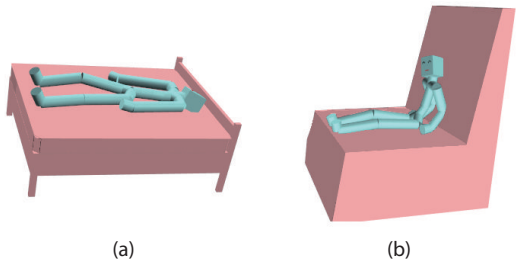

Figure 8: *The global shape of the IBS is also important: (a) and (b) are example scenes which produce similar multi-resolution feature vectors, while their IBS have different global shapes.*

4. Motion Comparison

The distance between two motions are computed by first extracting the key-frames, which are the representative frames of the entire motion (see § 4.1) and then applying Dynamic Time Warping (DTW) (see § 4.2).

4.1. Key-frame Extraction

We compute key-frames that are representative of the motion to cut the memory usage and computational cost when comparing the movements.

We make use of our representation for computing the key-frames such that they are based on the context of the scene. Previous methods, which are only based on the posture of the characters are not suitable when the scene involves close interactions between characters and objects. Here we use the distance function defined in the previous section to extract the frames that are different in terms of spatial relationships.

The key-frames are computed as follows: We first build a distance matrix as shown in Fig. 7 where the distances between each pair of configurations during a motion are recorded by using Eq. (4). The first two key-frames are the two frames with the maximum distance. To add another key-frame, we use the "farthest point strategy", which is finding the frame farthest (in other words, which has maximum distance) from existing key-frames. If this distance is greater than a threshold, which is set to 0.15 in our experiments, it is chosen as a key-frame. This process is repeated until no more

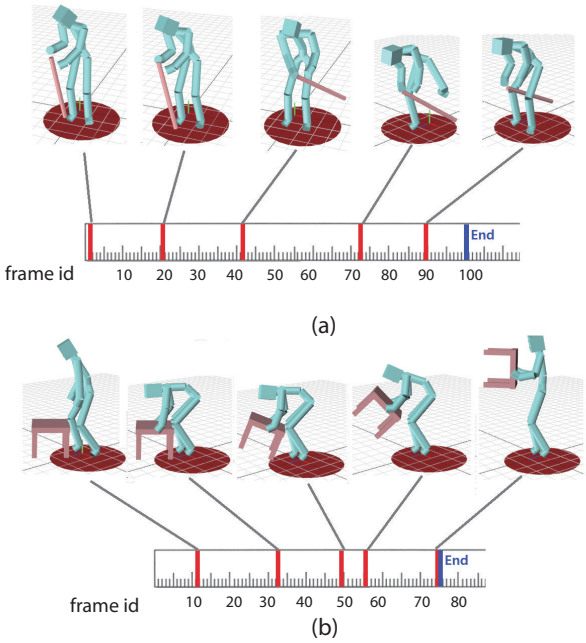

Figure 9: *Examples of the key frames for motion (a) "ride on the stick" and (b) "pick up the table".*

frames can be added to the key-frame list. Fig. 9 shows two series of key-frames computed by this method.

The key-frames are used as the representative frames of each interaction. When computing the distance between different interactions, we apply DTW, which will be explained in the next section, only to the key-frames rather than the entire motion, thus reducing the online costs and memory.

4.2. Dynamic Time Warping for Motion Comparison

Dynamic Time Warping (DTW) is a technique to align different time-series data, which is widely used for the alignment of speech, motion and video data. The key idea of DTW is to compute a cost matrix between every pair of input frames, and find a minimum cost path, which is typically located near the main diagonal of the matrix. The readers are referred to [MÃij07] for a more comprehensive explanation of DTW and its application.

We can compute the distance between two motions by applying DTW to the series of key-frames extracted using the method described in the previous section. Selecting the appropriate features and distance measure of the frames is the key of using DTW. In our method, the distance between frames is computed by Eq. (5). In the next section, we show motion retrieval results using this method.

5. Experiments and Results

In this section, we first describe the interaction dataset we use in our experiments (see § 5.1). Second, we briefly introduce other methods we compare our method with (see § 5.2). Then we show the results of single frame retrieval (see § 5.3), and lastly the results of motion retrieval (see § 5.4).

Object	Num of motions	Object	Num of motions
1. Large bin	19	8. Hula hoop	30
2. Small bin	22	9. Pistol	3
3. Box	25	10. Rifle	5
4. Chair	12	11. Ball	7
5. Table	25	12. Book	2
6. Broom	17	13. Cup	3
7. Tube	14	14. Hat	1

Table 1: *Interaction Motion List*

5.1. Dataset

The interaction data used in our experiments includes both the motion of the character and the object. We capture the motions by following the approach by Sandilands et al. [SCK13], where a magnetic motion capture system is used to track both the movements of the body parts and the object. The main advantage of using a magnetic motion capture system is that it does not suffer from occlusion problems. The system records both the translation and orientation of each sensor, so the configuration of each rigid bone/object can be recovered accurately by a single sensor.

We captured 164 short interaction motion clips between human and different objects. We put 14 sensors on the actor and 2 sensors on the object. We also used 21 motion clips captured by Sandilands et al. [SCK13], which can be downloaded from their project website. In total, there are 185 interactions with 14 types of objects. Table 1 lists the objects and the number of motions where the actor interacts with each object.

5.2. Alternative Methods

We compare our method to the following methods:

RELA-COOR: A state vector is formed by collecting the relative position of the object's center and all the body joints in the body root's coordinate system. The distance between different configurations are computed by the Euclidean distance between state vectors. Note that the pose of the body is also implicitly provided.

RELA-DIS: A state vector is formed by collecting the pairwise Euclidean distance between all the joint positions plus the object center. The distance between different configurations are computed by the L1 distance between two feature vectors. This method is based on [TLKS08] which uses pairwise Euclidean distance between joint pairs for motion comparison.

RELA-TRANS: A state vector is formed by computing a coordinate system using the axis of each bone and the object center, then computing the relative transformation between each pair of these local coordinates in the form of SE(3), and finally flattening it into a vector. The distance between different configurations are computed by the squared L2 distance between two features. This is an adaptation of the method proposed in [VAC14] to body-object interaction.

POS: A state vector is formed by collecting all the character's joint positions in the body root's coordinate system. The distance between two configurations is computed by summing the Euclidean distance between corresponding joints. This method is used as a baseline method to clarify the importance of encoding the relationship between body and object.

5.3. Retrieval by Single Frame

To examine the multi-resolution feature and the distance function we propose in § 3, we conduct a frame-based retrieval. Given a query frame, our system compares it to all the frames in the database and return the most similar frame in each motion, which corresponds to lowest distance. The returned frames are ranked according to the distance values. The query can be a frame from an existing interaction in the database or a scene designed by the user. In Fig. 10 we show the retrieval results for four example queries using our method.

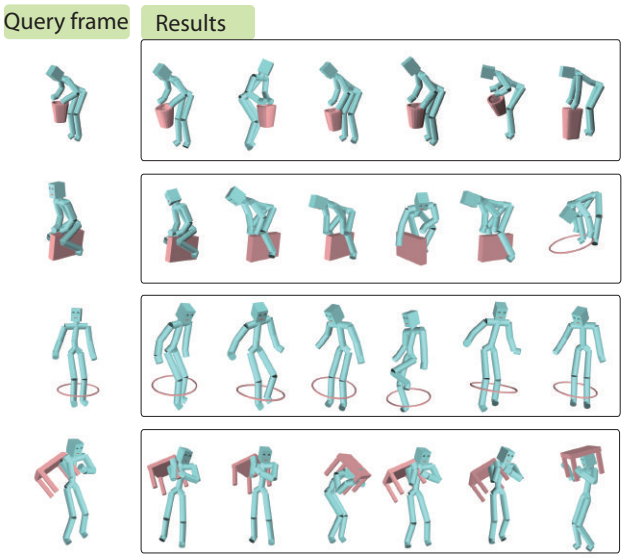

Figure 10: *Queries and retrieval results by using Eq. (5) (PFH+MULTI).*

Fig. 11 shows results of using different methods for frame retrieval. We compare the results of our method (PFH+MULTI) with others. The query is a frame where the character is sitting backwards on a chair. The context of this interaction is that the human character rides on an object. By using our method, the system can capture the context, and returns results where the character is riding on a chair or a box. The RELA-COOR method returns the frames where the object center is in front of the body. The RELA-DIS feature returns interactions where the object is in proximity to the legs. The method based on POS only returns scenes with similar postures. As the context of the scene is not explicitly encoded in POS, interactions where the character takes similar postures under different contexts are ranked high. To justify the design of our method, we also show the results when only using the MULTI feature or the PFH feature in Fig. 11. We can see that MULTI also only returns

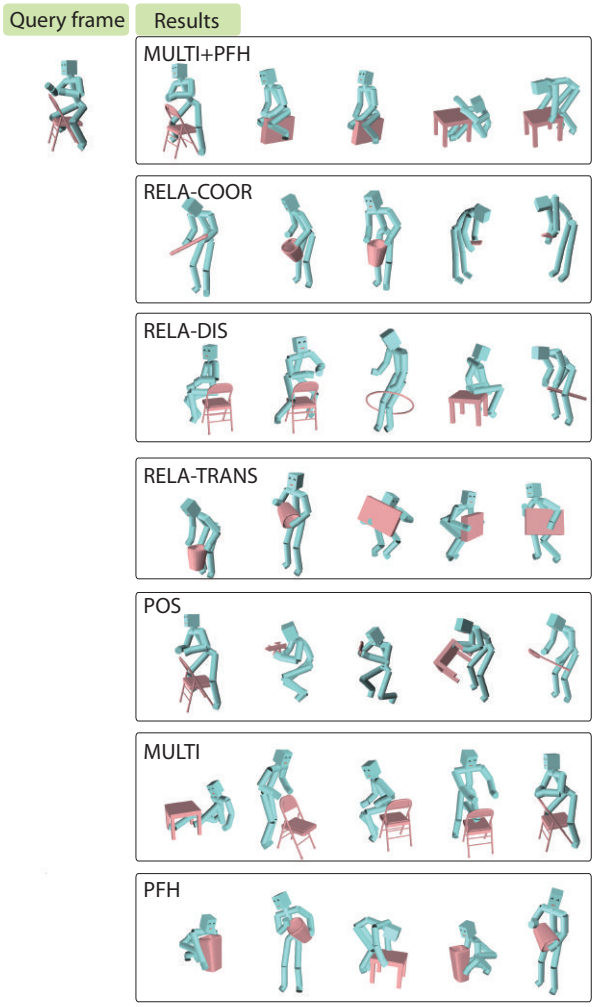

Figure 11: *Comparison of the retrieval results by using different methods.*

interactions where the objects are in proximity to the legs, while PFH returns the interactions where the object is half wrapped by the body, as these interactions produce IBS with similar shapes as the query. The results of MULTI+PFH is much better than PFH, because PFH does not consider which body parts are contributing to the interaction. The experiment results support our reasoning of PFH's limitation and show that the multi-resolution feature is an important supplement for describing character-object interactions.

Evaluation of Multi-resolution Distance

To justify the usage of the multi-resolution distance, we show the results of using different sets of resolutions for frame-based retrieval in Fig. 12.

From row 1 to row 4 in Fig. 12, we show the results of computing the multi-resolution distance when gradually increasing the

resolution. Considering the spatial relationship between the character and the object in the query, the most relevant results should be the character carrying an object on the right shoulder. From the results we can see that the most similar scene is moving to the top rank when we add in more level of details.

In the bottom row of Fig. 12, we also show the results of only using the finest level of details (the fourth level of segmentation). Although the two most similar "carrying table" frames also come to the top rank, the frames "carrying a hood on the shoulder" and "carrying a box when squatting" fail to appear due to the lack of information about the relationship that exists in the first three levels of segmentation. Without these upper level information, a small local change in the multi-resolution feature vector will result in a large distance, thus frames with similar relationships might be judged as very different. This result shows that we need to consider all different level-of-details when computing the distance between interactions.

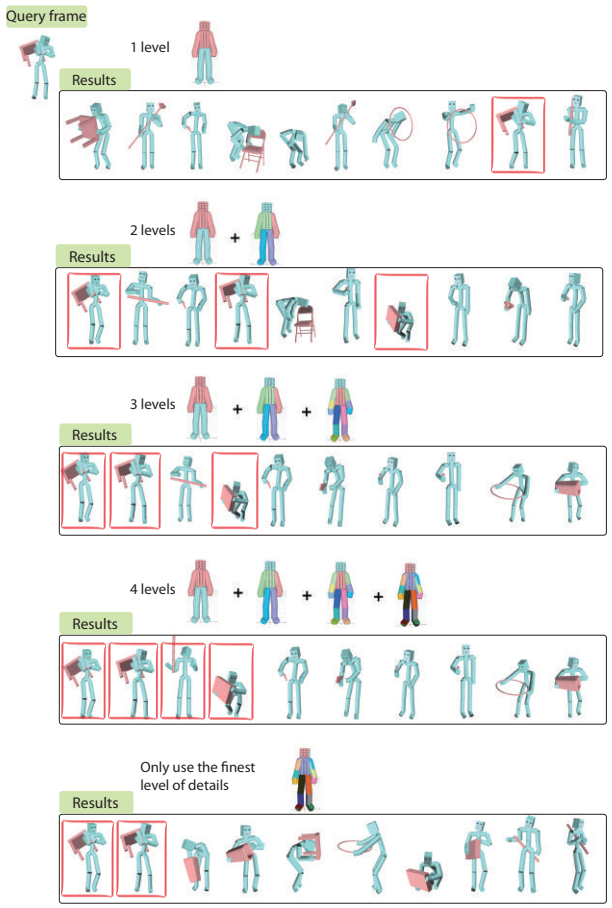

Figure 12: *The most similar frames retrieved when using different level settings for computing the distance. The results are sorted in the similarity order. The good results are highlighted with red borders. This figure aims to show how the retrieval results can be improved when higher level of details are considered.*

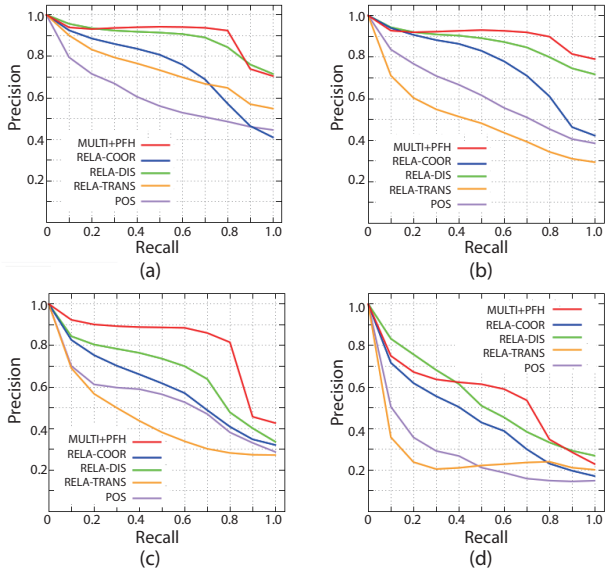

Figure 14: *Precision-Recall curves for (a) class 1: "pick up" motion, (b) class 2: "put down" motion, (c) class 3: "carry and move" motion and (d) class 4: "step over" motion.*

5.4. Retrieval by Motion

In this section, we retrieve interaction motions by using our method and other methods described in § 5.2. The retrieval results based on these features are compared and evaluated.

We manually classify 185 motion clips into the following four categories and use this classification as the ground truth:

- picking up an object (39 clips),
- putting down an object on the floor (33 clips),
- holding and manipulating an object (34 clips) and
- stepping over an object (18 clips).

Other 61 motions are labeled as "others" which are dissimilar from each other.

To show the resulting similarities of all motions in the database, we visualize the distance matrix by our method in Fig. 13 together with those by the other approaches. From Fig. 13 we can see that by using our method, the intra-class distance is generally smaller than the inter-class distance. It can be observed that the four classes are distinguished better by the MULTI+PFH feature than all the others.

To quantitatively compare the retrieval performances with other methods, we compute and show the average precision-recall curve of the four classes of motions when using the MULTI+PFH (our method), RELA-COOR, RELA-DIS, RELA-TRANS and POS method in Fig. 14. As MULTI+PFH encodes much richer information about the interaction between the body and the object, it performs better than other methods.

The main problem with the three relationship features RELA-COOR, RELA-DIS, RELA-TRANS is that the object is considered

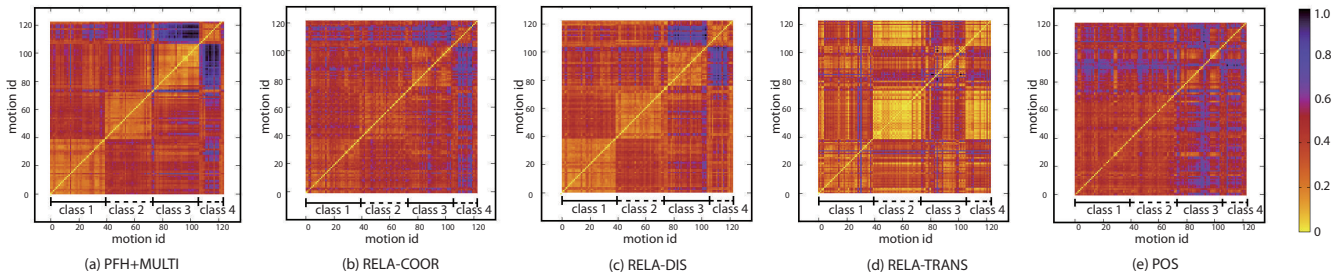

Figure 13: *Distance matrix of all the motions in our database, computed by MULTI+PFH, RELA-COOR, RELA-DIS, RELA-TRANS, and POS features. Compared to the other methods, the MULTI+PFH shows smaller intra-class distance than inter-class distance.*

as a point. Therefore, the shape information of the object is not encoded in these features, although the shape has a strong influence on the interaction. One way to improve the performance is to increase the number of sample points on the object. When taking such an approach, the objects must be aligned first and the sample points must be consistent between all the objects. In fact, we also did an experiment where we manually aligned the objects and sampled more points, but we found the performance drops due to the symmetry of objects. For example, grasping a handleless cup from different horizontal directions do not make any difference due to the symmetry. If the points are sampled in a way ignoring such symmetry, the feature vector can have different values even for equivalent movements. It is of course possible to do a symmetry analysis and change the way to sample points according to the symmetry, but this will require a complex shape analysis.

We can see that the RELA-DIS feature performs the second best, which shows computing distance between body joints, and body joint and object center is a reasonable way to encode the context of the interactions. However, in addition to the issue of the simplicity in representing the object by a point as mentioned earlier, the distance cannot encode the relative direction between the interacting parts. As a result, the RELA-DIS features may not be able to distinguish interactions such as "hold an object in front of the body" and "carry an object on the back". Our method can avoid such confusion by the finest level of segmentation of the body, where the front and back parts of the body have different labels. On the other hand, RELA-TRANS does not perform very well because it is designed for encoding the relationship between body parts but not between the body joints and the object.

Animated examples of the retrieval results by using our method are shown in the supplementary video.

6. Conclusion

In this paper, we propose a novel approach for computing the similarity between human-object interactions that takes into account the spatial-temporal relationship between them. We argue that using a representation that explicitly describes the spatial relationship between the body and objects can greatly improve the performance of indexing dynamic scenes. We develop a new multi-resolution distance function for the comparison of human-object interactions, and use it for motion retrieval. Through the experiments we show that our representation performs better than existing ones.

One advantage of using the IBS to encode the interaction is that we are relieved from analyzing the fine details of the object geometry, which is far more difficult than analyzing the IBS. Objects can have a wide variation in geometry and topology, which makes the computation of the correspondence not an easy problem. On the contrary, our method does not require computing such correspondence, but still takes into the geometry of the object through the IBS.

Our system is limited to the interaction between the body and a rigid object. Unlike [PKH*16], we do not provide a solution for interactions including deformable objects or fluid particles such as cloth, wind and liquid. Another limitation is that our system requires well defined object geometry. Missing parts of the objects would affect the accuracy of the interaction representation. However, capturing complex geometric surface during interaction is nontrivial.

In the future, we would like to apply our method on interaction data from different sources, such as RGB-D data captured by Kinect. Such a system will be useful for applications such as human action recognition, scene analysis and entertainment. It's also possible to apply our method for robot motion synthesis and planning. For example, we can learn a model of interaction from an existing interaction database. Then, given the geometry of the object, we can plan the motion of a robot by retargeting the human motion to that of the robot.

Acknowledgements

We thank the anonymous reviewers for their comments and suggestions. This work was supported by the China Postdoctoral Science Foundation (2015M582664), the National Science Foundation for Young Scholars of China (61602366), Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2016R1D1A1B03930472), and EPSRC Standard Grant (EP/H012338/1).

References

[BDH96] BARBER C. B., DOBKIN D. P., HUHDANPAA H.: The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical*

Software (TOMS) 22, 4 (1996), 469–483. URL: http://dl.acm.org/citation.cfm?id=235821. 3

[BSW05] BROWN M., SZELISKI R., WINDER S.: Multi-image matching using multi-scale oriented patches. In *Proc of IEEE CVPR* (2005), vol. 1, pp. 510–517. URL: http://dx.doi.org/10.1109/CVPR.2005.235, doi:10.1109/CVPR.2005.235. 4

[CCW*04] CHIU C.-Y., CHAO S.-P., WU M.-Y., YANG S.-N., LIN H.-C.: Content-based retrieval for human motion data. *Journal of Visual Communication and Image Representation 15*, 3 (Sept. 2004), 446–466. URL: http://www.sciencedirect.com/science/article/pii/S1047320304000331, doi:10.1016/j.jvcir.2004.04.004. 2

[CH05] CHAI J., HODGINS J. K.: Performance animation from low-dimensional control signals. *ACM TOG 24*, 3 (2005), 686–696. URL: http://doi.acm.org/10.1145/1186822.1073248, doi:10.1145/1186822.1073248. 2

[CLW*14] CHEN K., LAI Y.-K., WU Y.-X., MARTIN R., HU S.-M.: Automatic semantic modeling of indoor scenes from low-quality rgb-d data using contextual information. *ACM TOG 33*, 6 (2014). URL: http://doi.acm.org/10.1145/2661229.2661239, doi:10.1145/2661229.2661239. 2

[DGL09] DENG Z., GU Q., LI Q.: Perceptually consistent example-based human motion retrieval. In *Proc of I3D* (2009), ACM, pp. 191–198. URL: http://doi.acm.org/10.1145/1507149.1507181, doi:10.1145/1507149.1507181. 2

[FF05] FORBES K., FIUME E.: An Efficient Search Algorithm for Motion Data Using Weighted PCA. In *Proc of SCA* (2005), pp. 67–76. URL: http://doi.acm.org/10.1145/1073368.1073377, doi:10.1145/1073368.1073377. 2

[FH10] FISHER M., HANRAHAN P.: Context-based search for 3d models. *ACM TOG* (2010), 182:1–182:10. URL: http://doi.acm.org/10.1145/1866158.1866204, doi:10.1145/1866158.1866204. 2

[FRS*12] FISHER M., RITCHIE D., SAVVA M., FUNKHOUSER T. A., HANRAHAN P.: Example-based synthesis of 3d object arrangements. *ACM TOG 31*, 6 (2012), 135. URL: http://doi.acm.org/10.1145/2366145.2366154, doi:10.1145/2366145.2366154. 2

[FSH11] FISHER M., SAVVA M., HANRAHAN P.: Characterizing structural relationships in scenes using graph kernels. *ACM TOG 30*, 4 (2011), 34. 2

[GGVG11] GRABNER H., GALL J., VAN GOOL L.: What makes a chair a chair? In *Proc of IEEE CVPR* (June 2011), pp. 1529–1536. doi:10.1109/CVPR.2011.5995327. 2

[GMCL06] GAO Y., MA L., CHEN Y., LIU J.: Content-Based Human Motion Retrieval with Automatic Transition. In *Advances in Computer Graphics*, Nishita T., Peng Q., Seidel H.-P., (Eds.), no. 4035 in Lecture Notes in Computer Science. 2006, pp. 360–371. URL: http://link.springer.com/chapter/10.1007/11784203_31. 2

[HKT10] HO E. S., KOMURA T., TAI C.-L.: Spatial relationship preserving character motion adaptation. *ACM TOG 29* (2010), 33. URL: http://doi.acm.org/10.1145/1778765.1778770, doi:10.1145/1778765.1778770. 2

[HvKW*16] HU R., VAN KAICK O., WU B., HUANG H., SHAMIR A., ZHANG H.: Learning how objects function via co-analysis of interactions. *ACM TOG 35*, 4 (2016), 47:1–47:13. URL: http://doi.acm.org/10.1145/2897824.2925870, doi:10.1145/2897824.2925870. 2

[HZvK*15] HU R., ZHU C., VAN KAICK O., LIU L., SHAMIR A., ZHANG H.: Interaction Context (ICON): Towards a Geometric Functionality Descriptor. *ACM TOG 34* (2015). URL: http://people.scs.carleton.ca/~olivervankaick/pubs/icon.pdf. 2

[JKS13] JIANG Y., KOPPULA H., SAXENA A.: Hallucinated humans as the hidden context for labeling 3d scenes. In *Proc of IEEE CVPR* (2013), IEEE, pp. 2993–3000. URL: http://dx.doi.org/10.1109/CVPR.2013.385, doi:10.1109/CVPR.2013.385. 2

[KCGF14] KIM V. G., CHAUDHURI S., GUIBAS L., FUNKHOUSER T.: Shape2pose: Human-centric shape analysis. *ACM TOG 33*, 4 (2014), 120. URL: http://doi.acm.org/10.1145/2601097.2601117, doi:10.1145/2601097.2601117. 2

[KG04] KOVAR L., GLEICHER M.: Automated extraction and parameterization of motions in large data sets. *ACM TOG 23*, 3 (2004), 559–568. URL: http://doi.acm.org/10.1145/1015706.1015760, doi:10.1145/1015706.1015760. 2

[KPZ*04] KEOGH E., PALPANAS T., ZORDAN V. B., GUNOPULOS D., CARDLE M.: Indexing large human-motion databases. In *Proc of VLDB* (2004), pp. 780–791. URL: http://dl.acm.org/citation.cfm?id=1316689.1316757. 2

[LCK*14] LIU T., CHAUDHURI S., KIM V. G., HUANG Q.-X., MITRA N. J., FUNKHOUSER T.: Creating consistent scene graphs using a probabilistic grammar. *ACM TOG 33*, 6 (2014). URL: http://doi.acm.org/10.1145/2661229.2661243. 2

[LLLZ16] LI M., LEUNG H., LIU Z., ZHOU L.: 3d human motion retrieval using graph kernels based on adaptive graph construction. *Computers & Graphics 54* (Feb. 2016), 104–112. URL: http://linkinghub.elsevier.com/retrieve/pii/S0097849315001089, doi:10.1016/j.cag.2015.07.005. 2

[MRC05] MÃIJLLER M., RÃűDER T., CLAUSEN M.: Efficient Content-based Retrieval of Motion Capture Data. *ACM TOG* (2005), 677–685. URL: http://doi.acm.org/10.1145/1186822.1073247, doi:10.1145/1186822.1073247. 2

[MSSH14] MAJEROWICZ L., SHAMIR A., SHEFFER A., HOOS H. H.: Filling your shelves: Synthesizing diverse style-preserving artifact arrangements. *IEEE TVCG 20*, 11 (2014), 1507–1518. URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6636298. 2

[MÃij07] MÃIJLLER M.: DTW-Based Motion Comparison and Retrieval. In *Information Retrieval for Music and Motion*. Jan. 2007, pp. 211–226. URL: http://link.springer.com/chapter/10.1007/978-3-540-74048-3_10. 6

[PKH*16] PIRK S., KRS V., HU K., RAJASEKARAN S. D., KANG H., BENES B., YOSHIYASU Y., GUIBAS L. J.: Understanding and Exploiting Object Interaction Landscapes. *ArXiv e-prints* (Sept. 2016). arXiv:1609.08685. 2, 9

[RMB*08] RUSU R. B., MARTON Z. C., BLODOW N., BEETZ M., SYSTEMS I. A., MÃIJNCHEN T. U.: Persistent point feature histograms for 3d point clouds. In *In Proc. of Intelligent Autonomous Systems* (2008). 5

[SCH*14] SAVVA M., CHANG A. X., HANRAHAN P., FISHER M., NIE\S SNER M.: Scenegrok: Inferring action maps in 3d environments. *ACM TOG 33*, 6 (2014), 212. URL: http://dl.acm.org/citation.cfm?id=2661229.2661230. 2

[SCK13] SANDILANDS P., CHOI M. G., KOMURA T.: Interaction capture using magnetic sensors. *Computer Animation and Virtual Worlds 24*, 6 (2013), 527–538. URL: http://onlinelibrary.wiley.com/doi/10.1002/cav.1537/abstract, doi:10.1002/cav.1537. 6

[SKK04] SAKAMOTO Y., KURIYAMA S., KANEKO T.: Motion map: image-based retrieval and segmentation of motion data. In *Proc of SCA* (2004), pp. 259–266. URL: http://dl.acm.org/citation.cfm?id=1028557. 2

[SXZ*12] SHAO T., XU W., ZHOU K., WANG J., LI D., GUO B.: An interactive approach to semantic modeling of indoor scenes with an rgbd camera. *ACM TOG 31*, 6 (2012), 136:1–136:11. URL: http://doi.acm.org/10.1145/2366145.2366155, doi:10.1145/2366145.2366155. 2

[TCLK12] TANG J. K., CHAN J. C., LEUNG H., KOMURA T.:

Interaction Retrieval by Spacetime Proximity Graphs. *Comp. Graph. Forum 31*, 2 (May 2012), 745–754. URL: http://dx.doi.org/10.1111/j.1467-8659.2012.03033.x, doi:10.1111/j.1467-8659.2012.03033.x. 2

[TLKS08] TANG J. K., LEUNG H., KOMURA T., SHUM H. P.: Emulating human perception of motion similarity. *Computer Animation and Virtual Worlds 19*, 3-4 (2008), 211–221. URL: http://onlinelibrary.wiley.com/doi/10.1002/cav.260/abstract. 2, 6

[VAC14] VEMULAPALLI R., ARRATE F., CHELLAPPA R.: Human action recognition by representing 3d skeletons as points in a lie group. In *Proc of IEEE CVPR* (2014), pp. 588–595. URL: http://www.cv-foundation.org/openaccess/content_cvpr_2014/html/Vemulapalli_Human_Action_Recognition_2014_CVPR_paper.html. 2, 6

[WM95] WEBER J., MALIK J.: Robust computation of optical flow in a multi-scale differential framework. *International Journal of Computer Vision 14*, 1 (1995), 67–81. URL: http://dx.doi.org/10.1007/BF01421489, doi:10.1007/BF01421489. 4

[WXWL09] WU S., XIA S., WANG Z., LI C.: Efficient motion data indexing and retrieval with local similarity measure of motion strings. *The Visual Computer 25*, 5-7 (Mar. 2009), 499–508. URL: http://link.springer.com/article/10.1007/s00371-009-0345-1, doi:10.1007/s00371-009-0345-1. 2

[XMZ*14] XU K., MA R., ZHANG H., ZHU C., SHAMIR A., COHEN-OR D., HUANG H.: Organizing heterogeneous scene collection through contextual focal points. *ACM TOG 33*, 4 (2014). URL: http://doi.acm.org/10.1145/2601097.2601109, doi:10.1145/2601097.2601109. 2

[YYT*11] YU L.-F., YEUNG S. K., TANG C.-K., TERZOPOULOS D., CHAN T. F., OSHER S.: Make it home: automatic optimization of furniture arrangement. *ACM TOG 30*, 4 (2011), 86. URL: http://doi.acm.org/10.1145/2010324.1964981, doi:10.1145/2010324.1964981. 2

[ZHG*16] ZHAO X., HU R., GUERRERO P., MITRA N. J., KOMURA T.: Relationship templates for creating scene variations. *ACM TOG 35*, 6 (2016). URL: http://doi.acm.org/10.1145/2980179.2982410, doi:10.1145/2980179.2982410. 2

[ZWK14] ZHAO X., WANG H., KOMURA T.: Indexing 3d Scenes Using the Interaction Bisector Surface. *ACM TOG 33*, 3 (June 2014), 22:1–22:14. URL: http://doi.acm.org/10.1145/2574860, doi:10.1145/2574860. 2, 3, 4, 5