Computational Visual Media
https://doi.org/10.1007/s41095-021-0218-8 Vol. 7, No. 3, September 2021, 375–392
Research Article

Inferring object properties from human interaction and transferring them to new motions

Qian Zheng[1], Weikai Wu[1], Hanting Pan[1], Niloy Mitra[2], Daniel Cohen-Or[3], and Hui Huang[1] (✉)

Abstract Humans regularly interact with their surrounding objects. Such interactions often result in strongly correlated motions between humans and the interacting objects. We thus ask: "Is it possible to infer object properties from skeletal motion alone, even without seeing the interacting object itself?" In this paper, we present a fine-grained action recognition method that learns to *infer* such latent object properties from human interaction motion alone. This inference allows us to *disentangle* the motion from the object property and *transfer* object properties to a given motion. We collected a large number of videos and 3D skeletal motions of performing actors using an inertial motion capture device. We analyzed similar actions and learned subtle differences between them to reveal latent properties of the interacting objects. In particular, we learned to identify the interacting object, by estimating its weight, or its spillability. Our results clearly demonstrate that motions and interacting objects are highly correlated and that related object latent properties can be inferred from 3D skeleton sequences alone, leading to new synthesis possibilities for motions involving human interaction. Our dataset is available at `http://vcc.szu.edu.cn/research/2020/IT.html`.

Keywords human interaction motion; object property inference; motion analysis; motion synthesis

1 Introduction

Digitizing and understanding our physical world are important goals of both computer graphics and computer vision. In natural environments, humans regularly interact with surrounding objects, and such interactions result in strongly correlated motion between humans and these objects. Researchers in experimental psychology show that observers not only can recognize motion categories, but also *infer object properties* by observing corresponding human motion alone, even without directly seeing the object itself [1]. For example, we humans regularly estimate object properties like weight, spillability, path width, or shape, by observing either the real action of a human or even a pantomimed or virtual avatar action [2–4].

One way to computationally exploit such correlated human–object interaction motions would be to learn object properties by learning correlation with human skeletal motion over time. However, the available datasets for human activity recognition [5, 6] are RGB-D videos, which in general contain significant occlusions that hamper the extraction of unseen acting skeletons. While these videos can be used to broadly classify different actions [7], we still lack suitable datasets specifically designed for inferring fine-scale variations of object properties. Unlike previous work on action recognition, we analyze *similar actions* and hence have to learn subtle differences between actions of the same type that reveal latent properties of interacting objects. Inspired by previous work on motion style transfer, which transform an input motion into a new style while keeping its content, we use these latent properties to edit a given motion. For example, given the skeletal motion of a person walking on a wide path, we would like to synthesize the person's skeletal motion when walking on a narrow path.

In our work, we focus on eight typical types of human–object interaction, including lifting a box,

1 Shenzhen University, Shenzhen, China. E-mail: Q. Zheng, qianzheng85@gmail.com; W. Wu, wuweikai0617pk@gmail.com; H. Pan, panhanting95@gmail.com; H. Huang, hhzhiyan@gmail.com (✉).
2 University College London, London, UK. E-mail: n.mitra@cs.ucl.ac.uk.
3 Tel Aviv University, Tel-Aviv, Israel. E-mail: cohenor@gmail.com.
Manuscript received: 2021-01-22; accepted: 2021-02-24

Fig. 1 An actor is lifting a box from the table. Can skeletal motion tell us whether the box being lifted is light or heavy?

moving a bowl, and walking on a path. We collected video and 3D skeletal motions of actors using an inertial motion capture device, which do not suffer from the occlusion that is unavoidable in video-based recording. For these interactions, we learn to infer latent properties of the interacting object from the 3D skeleton sequences alone. In particular, we learn to identify the interacting object, by estimating its *property values*, such as 0 kg, 15 kg, or 25 kg for box weight, or empty or full for bowl spillability.

For the inference task, we treat object latent property estimation as a fine-grained classification problem by analyzing similar input skeletal motions. Although some properties (e.g., the weight) may vary continuously, treating it as a regression problem requires more training samples. We represent a skeleton sequence as a time sequence of graph structures, which encodes the position and speed information of all joints with temporal dynamics. After analyzing per-joint features, we feed it into a recurrent network to recognize the latent object properties. The results obtained demonstrate that the interaction motions and interacting objects are highly correlated, allowing object property values to indeed be inferred, to a certain accuracy, by just observing human movements. We will show that, in comparison to existing works on action recognition, our method achieves higher inferencing accuracy.

For the synthesis task, we develop a network architecture to disentangle object properties from the abstract motion, which allows us to create novel skeletal motions by mixing new object properties with target skeletons. We train a deep neural network with a simple encoder–decoder structure to perform the disentanglement, i.e., the latent space encodes the motion content *without* object property. A motion can then be synthesized given a specific property value.

In summary, we claim the following contributions:

- learning subtle differences between motions of the same type, of humans interacting with an object,
- a property and motion disentanglement network that allows motion synthesis conditioned on target interactions, and
- a public, extensive, interaction dataset for inferring object properties from motions with 4k+ samples collected from 100 participants, including eight everyday interactions: lifting a box, moving a bowl, walking, fishing, pouring liquid, bending, sitting, and drinking.

2 Related work

Our work analyzes human interaction motion to detect object properties. Therefore, we briefly describe previous approaches that exploit human–object interaction in visual inputs, with a focus on object property inference. Since we use skeleton sequences to represent motions, we also review related works on skeleton-based action recognition.

2.1 Human–object interaction

Human–object interaction detection is an important scientific problem [8] with wide practical uses. Recent methods can successfully detect <human, verb, object> triplets from visual inputs [9, 10].

A variety of techniques in shape analysis have been developed to extract functional information about objects and scenes using human–object interaction as cues. An appropriate human pose or action map can be created from an input object [11–13] or scene [14, 15]; see the survey in Ref. [16] for more information.

Hidden human context has been used as a cue for labeling and arranging scenes [17, 18]. However, there is no work yet solving the inverse problem: *inferring object properties from human motions and/or interactions alone.*

The spatial relationship between the characters and objects in the environment captures the semantics of interactions. Ho et al. [19] introduced an interaction mesh structure to explicitly represent this spatial relationship for motion retargeting. Later this representation was used for motion comparison [20].

2.2 Object property inference

Researchers in psychology have reported that observers can make fine distinctions when presented with human motions in visual form. The weight of a box can be *seen* by observing another person lifting and carrying it [2], and the elasticity of a supporting surface can be judged by observing a person walking on it [21]. Vaina et al. [4] demonstrated that the weight of an object can be robustly estimated, while size and shape are harder for observers to estimate. Recently, Podda et al. [3] showed that participants are able to identify the weight of a grasped object from both occluded real and pantomimed movements, solely using available kinematic information. Observers seem to focus most on the duration of the lifting movement to perceptually judge weight [22]. Some findings suggest observers may integrate multiple sources for object property inference; for example, shape, motion, and optical cues are used when inferring stiffness [23]. Still, we focus on inference from motion alone in this work.

Koppula et al. [24] proposed a method to learn human activities from RGB-D video by jointly modeling human sub-activities and associated object affordances. Object classes and their 3D locations can be recovered from motion by exploiting human–object spatial relations, and used for synthetic scene reconstruction [25] and scene arrangement recovery [26]. There has been little effort to automatically infer other properties. Davis and Gao [27] presented a computational framework that can label the effort of an action corresponding to the perceived level of exertion by the performer. Gupta and Davis [28] classified objects as heavy or light based on the velocity of ballistic motions detected in video. Integrating a 3D physics engine is another way to infer physical properties, including mass, position, 3D shape, and friction, from real-world videos [29, 30].

2.3 Action recognition and motion style transfer

With the availability of large-scale skeleton datasets, deep learning is popular for action recognition. Skeleton sequences are time series of joint positions. Recurrent neural networks, designed to model long-term temporal dependency problems, have been well exploited for skeleton sequences [31–33]. The skeleton is also a special graph structure representation, and thus graph convolution networks can be utilized as well for action recognition [34].

CNN models are able to extract high-level information and have also been used to deal with skeleton sequences. A skeleton sequence can be converted into an image or a 3D tensor, and then fed into a CNN to recognize the underlying action. These methods vary most in the representations of skeleton sequences and network structures. Ke et al. [35] represented a skeleton sequence as several images to encode different spatial relationships between joints, and then applied a pre-trained VGG to extract the features. Li et al. [36] represented a skeleton sequence as a 3D tensor, and modeled global co-occurrence patterns with a CNN. Most recently, Aristidou et al. [37] used a triplet loss network to map short motion clips to an embedding space, where the distances represent similarity between motion clips. We also utilize graph convolution and an RNN to learn object properties from skeletal motions. Nonetheless, we use sub-categorical properties to effectively distinguish fine-grained differences between motions of the same class.

Another related topic is motion style, which usually represents the mood or identity of a particular character's motion. By analyzing differences between performances of the same content in different styles, researchers have proposed methods to transform an input motion data into new styles [38–40]. Bellini et al. [41] introduced a method to enhance the rhythm of a dancing performance video by detecting motion beats and synchronizing them with music beats. Object properties and actions are significantly correlated. A particular object property can be only observable in a particular action type, which makes the existing motion style transfer techniques unsuitable for our synthesis task.

3 Interaction motion dataset collection

3.1 Considerations

Traditionally, human motion is captured using optical marker-based systems with markers placed on the performer. With recent success of deep learning, 2D poses [42–46] and 3D poses [47–52] can be extracted directly from RGB or RGB-D video sequences. Large-scale skeletal motion datasets, such as CMU [53], NTU RGB+D [5], and PKU-MMD [6], are available as driving forces for motion recognition, retrieval, and synthesis. However, although these datasets contain human–object interaction motions, the object information is usually unlabeled, and the (partial) joint trajectories are insufficient to reliably infer 3D object properties. For example, some limbs are very likely to be occluded by the interacting objects. Such occlusion makes it very difficult to robustly extract high-quality skeletal motions from monocular or RGB-D videos, even with state-of-the-art pose detection methods. This is particularly true in our setting where we seek subtle motion differences. Therefore, we use inertial measurement units (IMUs) to obtain 3D human motions that are totally occlusion free.

3.2 Data modalities

We utilized multiple data modalities to construct our dataset. When performing actions, each subject wore an Xsens MVN inertial motion tracking suit to capture high-quality 3D skeleton information at 240 frames per second. Each subject was also required to wear a head-mounted camera to capture ego-centric video. Further, we used three uncalibrated cameras to record the subject from three different views, storing three videos at 50 frames per second. For each interacting object, in addition to measuring its size and weight, we also scanned its geometric shape. Figure 2 presents our capturing scenario and the data modalities of each motion sample collected. Although in this work we only use 3D skeletal information to infer object properties, we believe that these data modalities will be useful for future research.

3.3 Subjects and object interactions

We carefully selected human–object interactions to depict the correlation between human motions and properties of objects. For a good candidate, object property values could be inferred easily from the whole interaction motion alone, but only with difficultly from a single static frame. Following this rule, we chose eight daily interactions: Walking for estimating *the width* of the path, Fishing for *the length* of a fishing rod, Pouring for *the type* of liquid, Bending for *the stiffness* of a power twister, Sitting for estimating *the softness* of a chair being sat on, Drinking for estimating *the amount* of water inside a cup, Lifting a box for *the weight* of an object being lifted, and Moving a bowl for *the spillability* of an object. These motions are shown in Fig. 3. We used 100 different subjects during data collection. They varied in age (20–35), gender (M or F), height (150–195 cm), and strength (weak–strong). Here we briefly describe the setting of Walking; see the Appendix for other interactions.

Walking. Each subject was asked to walk back and forth on three straight paths of different widths. We delimited the width of a path using line markers, and asked the subjects to not cross the edges. This gave a total of $3 \times 2 \times 100 = 600$ motion samples.

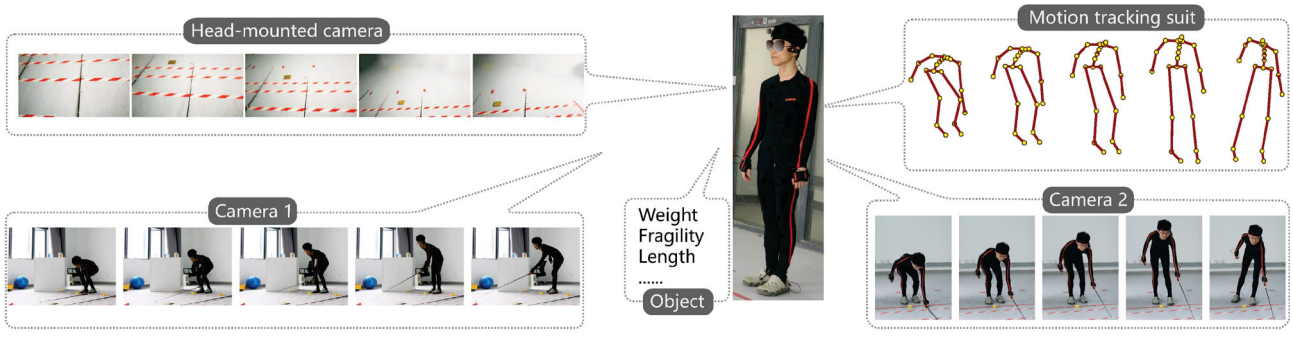


Fig. 2 For each sample, we capture a 3D skeleton sequence using an inertial motion tracking suit, an ego-centric video by a head-mounted camera, two other videos by two cameras placed outside, and the object's geometry along with its properties.

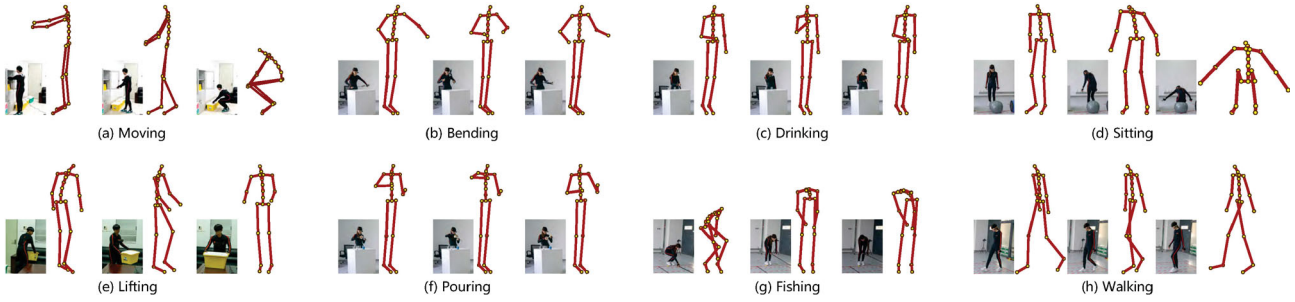


Fig. 3 Eight interaction motions represented in our dataset, which contains 4k+ interaction captures across 100 different participants.

4 Object property inference

4.1 Skeleton sequence representation

The input skeleton data is a sequence of multi-frame tree structures with 3D joints as nodes that form an *action*. As shown in Fig. 4, a skeleton sequence is represented by a 3D tensor of size $T \times J \times D$, with T representing the frame length, $J = 23$ the total number of joints, and D the feature dimension of each joint, respectively.

Representing a skeleton sequence by joints in xyz locations is common [5, 35, 36]. Some researchers also represent the joints using 3D angles [37]. In our case, the object properties that we aim to estimate are highly correlated with the dynamic properties of motions. As we show later, joint trajectories (position and velocity representations) can overall help with object property inferencing.

Each joint is represented by the x, y, and z coordinates in a local body coordinate system with its origin on the pelvis (indicated by a blue dot in Fig. 4). The local coordinate frame has the z axis perpendicular to the floor, and the x axis parallel to the 3D vector from the right shoulder to the left shoulder. For each frame, we use the xyz position relative to the current pelvis joint. Note that in this representation, we ignore the movement of the pelvis in the sequence. We also explicitly encode the velocities of joints. Let the ith joint's position in frame t be J_i^t. Then, the velocity of joint S_i^t is approximated by the temporal difference between two consecutive frames:

$$S_i^t = (J_i^{t+1} - J_i^t)/\delta t$$

where δt represents the corresponding time interval.

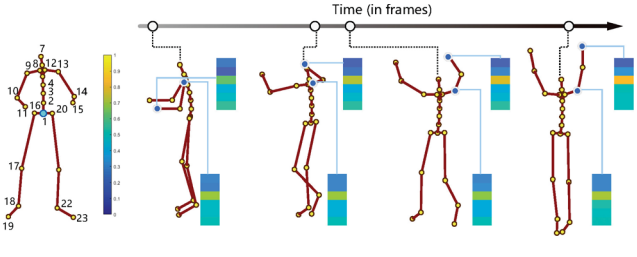


Fig. 4 We represent a skeleton sequence as a tree sequence. The input feature of each joint is represented by its xyz location and velocity in local body frame coordinates. The cyan point indicates the root (pelvis) of the tree. Each block indicates the joint's feature in a frame.

4.2 Object property classifier

4.2.1 Overview

In practice, our object property classifier consists of two graph convolution layers, a GRU layer [54], and then two fully connected (FC) layers for the final classification, i.e., making the object property inference; see Fig. 5. The graph convolution layer computes per-joint features taking into account the known human body skeleton topology. The GRU layer with attention accumulates the information from all frames and computes the importance of each joint. The combination of graph convolution layers and GRU units enables us to better infer object property values from the same types of motions.

4.2.2 Graph convolution layer

Graph convolution usually deals with an undirected graph. As the skeleton is a hierarchical tree structure,

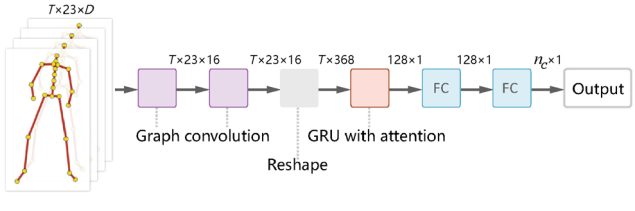


Fig. 5 We represent a skeleton sequence by a 3D tensor of size $T \times J \times D$, T representing the frame length, J the number of joints, and D the feature dimension of each joint. Our classifier for object property values is made of graph convolution layers, GRU, and fully connected layers. The size of the tensor after each layer is indicated in the figure with n_C denoting the number of classes for an object property, e.g., $n_C = 6$ when the input is a lifting motion and the object property is the weight of box being lifted.

for a given joint, we only consider its parent, instead of all neighbors, to apply a convolution. Formally, for the ith joint of frame t, its feature after graph convolution $\boldsymbol{x}'_{t,i}$ is

$$\boldsymbol{x}'_{t,i} = \text{Relu}\left(\boldsymbol{W}_g \begin{bmatrix} \boldsymbol{x}_{t,i} \\ \boldsymbol{x}_{t,j} - \boldsymbol{x}_{t,i} \end{bmatrix} + \boldsymbol{b}_g\right) \quad (1)$$

where $\boldsymbol{x}_{t,i}$ represents the feature of this joint fed to this layer, j is its parent's index, and $\boldsymbol{W}_g, \boldsymbol{b}_g$ are the learnable weights for a graph convolution layer. Experiments clearly show that using skeleton topology information can improve inferencing accuracy; see, e.g., Fig. 10. We use this asymmetric edge function as suggested in Ref. [55].

4.2.3 GRU layer with attention

Attention mechanics is widely used in skeleton-based action recognition. It can improve action recognition and discover the relative importance of joints and frames. For example, Zhang et al. [56] used an element-wise attention gate to an RNN block to improve action recognition. We also add a joint-wise gate to the RNN cell. The attention value of each joint in frame t is computed based on the hidden state of the RNN cell $\boldsymbol{H}_{t-1}$:

$$a_{t,i} = \text{sigmoid}(\boldsymbol{W}_h \boldsymbol{H}_{t-1} + \boldsymbol{W}_x \boldsymbol{x}_{t,i} + \boldsymbol{b}_a) \quad (2)$$

where $\boldsymbol{x}_{t,i}$ represents the feature of the ith joint fed to the RNN cell, and $\boldsymbol{W}_h, \boldsymbol{W}_x, \boldsymbol{b}_a$ are the learnable weights for an attention convolution layer. Then, the input fed to the RNN cell is updated using $\tilde{\boldsymbol{x}}_{t,i} = (1 + a_{t,i})\boldsymbol{x}_{t,i}$, where $a_{t,i}$ represents the importance of the ith joint in frame t.

4.2.4 Implementation details

For all experiments presented here, we use $J = 23$ major body joints. We use classical cross entropy loss as we have a classification problem. For skeletal representation, we apply a normalization pre-processing step. The lengths of collected motion samples vary from 3 to 6 s. Additionally, we used data augmentation to increase the number of samples and to remove rotation bias. We rotated each sequence along the z axis 10 times and cropped 10 sub-sequences from each original and rotated sequence. The rotation angles were drawn from a uniform distribution between $[0, \pi)$, and the cropping ratios were drawn from a uniform distribution $U[0.9, 1]$. This data augmentation enlarged the size of our skeletal motion dataset 100 times. We down-sampled each sub-sequence to 30 frames. We used TensorFlow with the network initialized using the Adam optimizer with a batch size of 32 and a learning rate of 0.0001. Training was stopped after 60 epochs by default.

5 Object property-aware motion transfer

In synthesis, our goal is to use target object property values to guide motion transfer for a given actor. Given an interaction skeletal motion x whose object property value is y, and a new target object property value y', we want to generate new skeletal motion x' that matches the given target property value y'.

Inspired by Refs. [57, 58], we use an encoder–decoder structure to perform this motion retargeting; see Fig. 6. The encoder E converts an input motion to a latent space $z = E(x)$, and the decoder D synthesizes a new motion conditioned on the target property value, denoted $D(E(x), y')$. To train the network, we use a loss function consisting of two terms: a reconstruction loss and a contrastive loss.

The *reconstruction loss* aims to constrain the encoder and decoder. We want the output motion to be similar to the motion performed by the same subject under the target property value y', denoted by $\hat{x}$. When y' equals y, $\hat{x}$ equals x. We use the Euclidean loss in the local coordinate frame to measure the quality of the reconstruction:

$$\mathcal{L}_{\text{rec}}(E, D) = \mathbb{E}_{x,y'} \|D(E(x), y') - \hat{x}\|_2^2 \quad (3)$$

The exact choice of the reconstruction loss is not fundamental here. Other formulations of reconstruction loss especially designed for motion frames, such as geodesic loss measuring the 3D rotation errors of joints [59], could be used.

Another loss is a *contrastive loss* that ensures that $E(x)$ does not have residual information about the input object property [60]:

$$\mathcal{L}_{\text{ctr}}(E) = \mathbb{E}_{x,x^+} \|E(x) - E(x^+)\|_2^2 + \mathbb{E}_{x,x^-} \left[\alpha - \|E(x) - E(x^-)\|_2\right]_+^2 \quad (4)$$

To help disentanglement, we constrain the distance in latent space between different motion samples. Taking an anchor motion x, we compare it with a positive motion x^+ that comes from the same performer under a *different* object property value, and a negative motion x^- coming from a different performer under the *same* property value. The dissimilarity between the anchor motion and negative motion should be larger than a margin α, and the distance between the anchor motion and positive

motion should be small. The full objective function to optimize the encoder E and decoder D is a combination of two terms:

$$\mathcal{L}(E,D) = \mathcal{L}_{\text{rec}}(E,D) + \lambda \mathcal{L}_{\text{ctr}}(E) \quad (5)$$

where λ is a hyper-parameter that controls the relative importance of contrastive loss compared to reconstruction loss. We use $\lambda = 0.1, \alpha = 5$ in all our experiments.

Here the skeleton sequence for motion transfer is represented by local and global motion as suggested in Ref. [57], which is slightly different from that for object property inference. For local motion, we use joints in xyz locations of local frame coordinates, just as for property inferencing. Global motion consists of the root's global velocity and foot contact labels. See Fig. 6; the rows represent the location of a joint over time. We down-sample the motion to 64 frames.

The encoder is composed of 4 1D convolutional layers with stride size two for down-sampling the time axis. The decoder is composed of 4 nearest-neighborhood up-sampling followed by convolution with stride 1 to restore the motion; see Fig. 6.

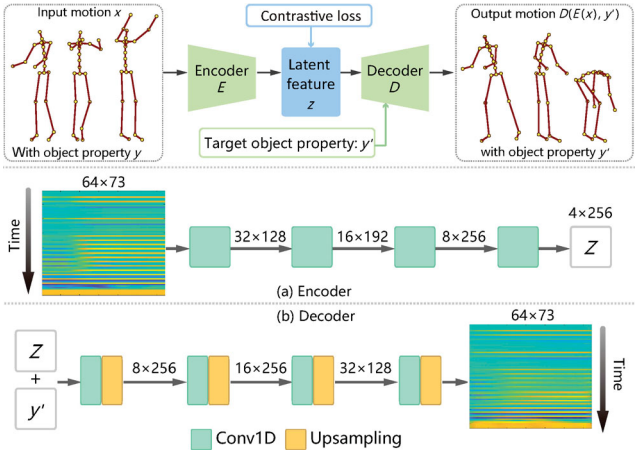


Fig. 6 Network for motion transfer driven by object properties. By changing the object property value y, we may generate human motions that match the given property value well.

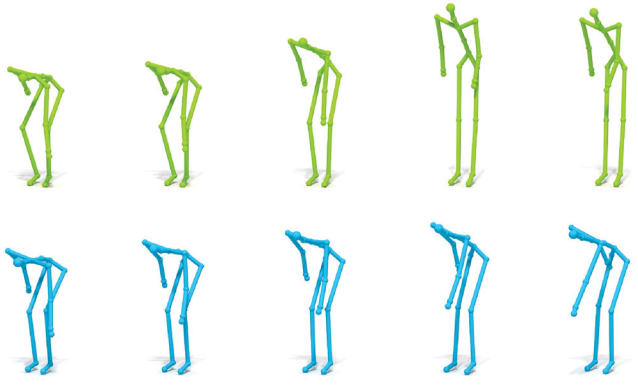


Fig. 7 Given a fishing motion with a long rod (green), we transfer the rod from *long* to *short* to get a new motion (blue).

All models were trained using Adam with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The batch size was set to 32 for all experiments. We trained all models with a learning rate of 0.00001. Training takes about 10 minutes on a server with an Intel Xeon 2.20 GHz CPU 10 cores, 256 GB memory, and a NVIDIA TitanXP GPU.

6 Results and evaluation

6.1 Evaluation of object property inferencing

6.1.1 Basis

To measure model performance on object property inferencing, we conducted a cross-subject evaluation. We split the 100 participants into training (60), validation (20), and testing (20) groups, respectively. Thus, testing is done with different people to the ones who were employed for training and validation. During training, we selected the network parameters with the smallest validation error over all iterations. Then, we evaluated and reported performance on the testing group.

We implemented several variants to evaluate the impact of different skeleton representations. As using both position and speed achieves the best performance, we applied this representation in other tests. We report the object property inferencing accuracy on all eight types of motion. To evaluate, to set a baseline, we used a state-of-the-art method for action recognition based on skeletons. We also evaluated the utility of the graph convolution layer and GRU units with attention. Furthermore, we tested the inferencing accuracy regarding the sensitivity to object property differences.

Table 1 shows the object property inferencing accuracy (%) for the cross-subject settings. The performance looks unimpressive at first glance. Nonetheless, in consideration of the subtle differences between motions under different object properties, we believe this accuracy is reasonable. Furthermore, in most cases, our method outperforms the baseline. We describe lifting motion in detail in the following as an example. We wish to estimate weight from a lifting motion. We trained a classifier that outputs 6 classes corresponding to weights from 0 to 25 kg in

Table 1 Object property inferencing accuracy (%) on the cross-subject settings. Weight has 6 classes (0, 5, 10, 15, 20, and 25 kg). Spillability has 3 levels, reflected by moving without spilling an empty bowl, a bowl full of rice, and a bowl full of water. The width of the path, length of the rod, type of the liquid, stiffness of the power twister, and water amount in the cup also have 3 levels. The softness of the chair has 4 classes

Object property	Accuracy (%)	
	Ours	ST-GCN
Lifting a box for weight (6)	**61.8**	57.3
Moving a bowl for spillability (3)	77.5	**78.9**
Walking for path width (3)	**83.9**	73.8
Fishing for length of rod (3)	**80.7**	77.2
Pouring for type of liquid (3)	**62.8**	62.1
Bending for stiffness (3)	**71.6**	44.7
Sitting for softness of chair (4)	**73.7**	66.4
Drinking for amount of water inside a cup (3)	**62.5**	57.0

steps of 5 kg. The accuracy is about 62% on the cross-subject setting. As the weight differences between the classes are relatively small and the lifting motion is also highly related to the strength of the performer, the resulting estimation accuracy is acceptable for such subtle changes.

6.1.2 Baseline

We used a state-of-the-art method for action recognition based on skeletons [34] (denoted ST-GCN) as baseline to evaluate fine-grained motion inferencing. ST-GCN consists of 9 layers and has about 0.3 million parameters, about ten times more than our model. The original network performed very poorly probably due to the small size of our motion dataset. Setting the number of layers to three achieved the best performance during tuning. We thus reduced the original ST-GCN to three layers. This also leads to a similar parameter setting to ours. We also used both position and speed to represent the skeletal motion. The last column in Table 1 shows performance in the cross-subject setting. Overall, our proposed method has achieved higher inferencing accuracy.

6.1.3 Choice of skeleton representation

To evaluate the impact of skeleton representations, we tried several variants. A skeleton sequence was represented by the positions of joints, or the orientation of bones. Similarly, motion dynamics were measured by joint speeds or bone angular speeds. We represented the skeleton sequence in different forms, and then evaluated their performance on object property inferencing for three different motions: lifting, walking, fishing. All other settings were exactly the same. Table 2 shows that the best representation varies for different object properties. Overall, using both position and speed is a good option, so this representation was used in other experiments.

Table 2 Impact of different skeleton representations on inferencing accuracy (%) in the cross-subject setting

	Lifting (6)	Walking (3)	Fishing (3)
Position	57.82	76.84	**84.21**
Euler angles	43.38	81.58	73.68
Speed	59.93	79.82	69.4
Angular speed	47.46	73.16	63.51
Position, Euler angles	55.70	79.65	71.58
Position, speed	**61.81**	**83.93**	**80.70**
Position, angular speed	**64.58**	79.47	77.54
Speed, angular speed	55.70	**84.39**	76.49
Speed, Euler angles	50.56	70.00	66.67
Euler angles, angular speed	56.06	80.53	72.28
Position, Euler angles, angular speed	50.35	78.42	70.18
Position, speed, angular speed	**62.32**	**82.98**	**78.95**
Position, Euler angles, speed	56.55	81.58	71.93
Position, Euler angles, speed, angular speed	58.73	82.98	78.95

6.1.4 Graph convolution

To evaluate the impact of the graph convolution layer regarding per joint features, we fixed other layers and only changed the two graph convolution layers, and reported the performance on object property inferencing; see Fig. 10. We evaluated in different settings: ignoring the connections between joints and only considering each joint itself to compute per joint features (as in PointNet [61]), or treating the skeleton as a tree whose root is the pelvis (a directed graph), or treating it as an undirected graph. We also considered different numbers of ancestors (from 1 to 3) of each joint. For an undirected graph, we also considered its k-degree neighborhoods using $k = 1, 2, 3$, or all nodes (FC-Graph) in our tests. Figure 10 shows that though the inferencing performance varies across the types of motions, considering a joint's parent to compute its feature is a good option.

6.1.5 Joint-level attention

The learned attentions marginally improved object property inferencing, especially for rod length inferencing in Fishing and softness of chair inferencing from Sitting: both increased by about 4%. We visualized the attention weights on joints by color. For better visualization, we linearly mapped the squared attention values to colors to highlight their importance. Figure 8 shows the attention weights

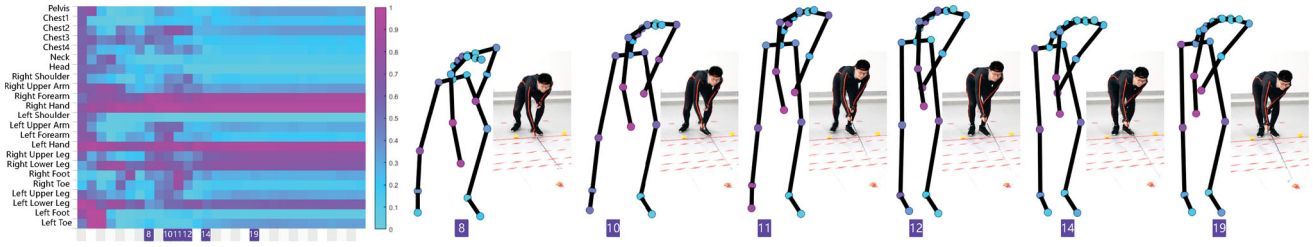


Fig. 8 Estimating the joint-level importance of a fishing motion for inferring the object property. Note that here the color of magenta to cyan indicates the importance from high to low.

on the two arms are large for the *fishing* motion, consistent with our human intuition.

6.1.6 Weight and water amount sensitivity

To evaluate inferencing accuracy with respect to sensitivity to object property differences, we trained and tested the model with several different subsets of motion samples, i.e., using samples with only certain specific property values. For example, when evaluating the model's ability to distinguish 5 kg from 10 kg, only motion samples with these two weights were used. All other settings were exactly the same.

Table 3 shows that inferencing performance is related to the weight label distribution. Note that 2-class classification accuracy drops dramatically from 94.7 down to 78.7 when classifying 10 and 15 kg boxes instead of 5 and 25 kg, even lower than the 3-class classification accuracy when classifying 5, 15, and 25 kg. We argue that this is mainly caused by the small dynamic motion differences when lifting boxes are close in weight. The water amount label distribution also shows a similar trend.

Table 3 Object weight and water amount inferencing accuracy (%) doe different configurations: two, three, or six classes

Lifting (kg)				Drinking	
5/25 (2)	10/15 (2)	5/15/25 (3)	(6)	Empty/full (2)	(3)
94.7	78.7	81.7	61.8	86.8	62.5

6.2 Video comparison

6.2.1 Property inferencing only from video

We additionally evaluated the weight and spillability inferencing performance using different input sources. In particular, we tested the performance using 2D skeleton sequences directly extracted from videos that were recorded from a fixed view. We used OpenPose detector [42] to extract 25 body keypoints in 2D to get image-space skeletons using videos. Due to the fixed camera view and the occlusion of interacting objects, extracted 2D skeletons may have large missing parts in some frames; see, e.g., Fig. 9(top). We chose the most representative 17 body joints, and replaced the 3D IMU skeletons with corresponding 2D video skeletons. Now the skeleton sequences have only x and y positions without z coordinates. The speed and acceleration attributes are not used as there are unavoidable flickers in video sequences and they cannot be easily lifted to 3D.

Figure 11 presents the evaluation of 6-class weight classification and 3-class spillability inferencing on

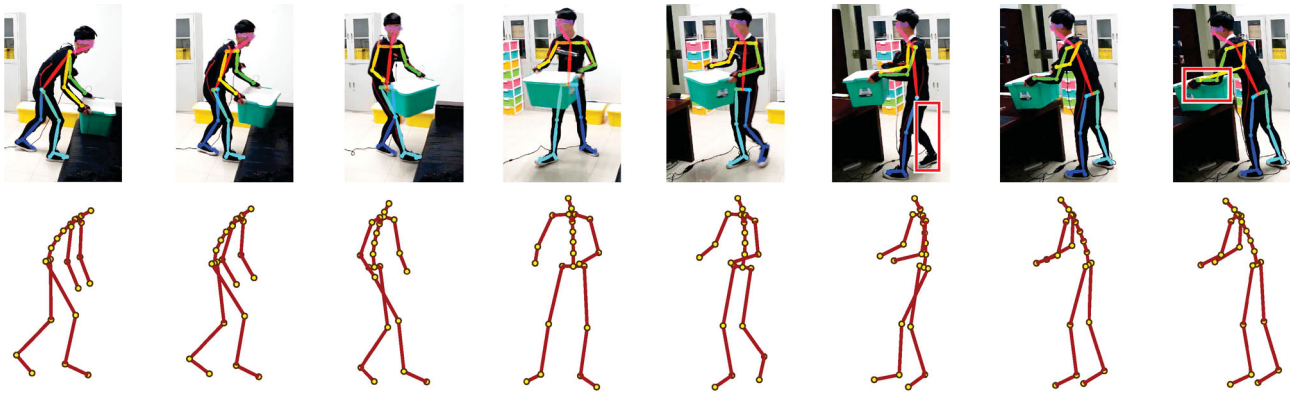


Fig. 9 We show some 2D skeletons extracted from our recorded video at the top, where missing parts are highlighted with red boxes. In comparison, 3D IMU skeletons captured at the corresponding frames are shown underneath, which are clean and complete.

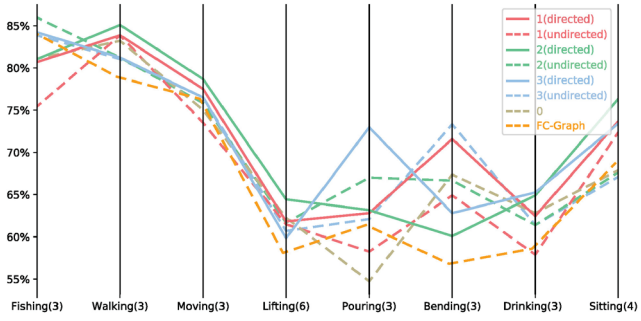


Fig. 10 Parallel coordinate representation of inferencing accuracy for different ways of computing per-joint features in the two graph convolution layers. Each vertical axis represents the inferencing accuracy for a type of motion. Each line represents a setting. Considering all motion types, it seems good to use the parent of a joint to compute joint features (red solid line).

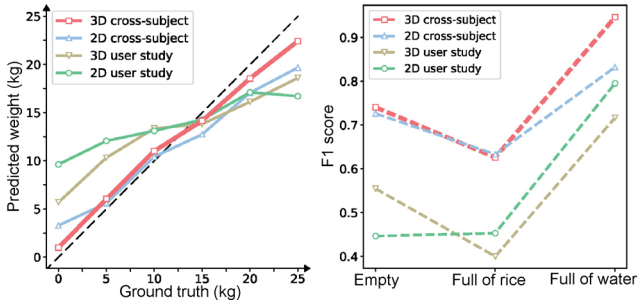


Fig. 11 Left: average predicted weights by our model and human observers for both 2D and 3D skeletal cases, where the weights are one of 0, 5, 10, 15, 20, and 25 kg. The 6-class weight predictions by our model using 3D skeletons are much closer to ground truth indicated by the slanting black line. Right: the F1 score per class of the spillability estimate by our model and human observers. For both 2D and 3D skeletal cases, our method (see red and blue marks) achieves better results.

cross-subject settings, using our model trained on 2D and 3D skeletons and human observers. Using 2D skeletons instead of 3D causes some drop in inferencing accuracy in both weight and spillability estimation: see the red and blue lines. We believe this is mainly due to joint estimation errors, missing depth information, and kinematic flickering artifacts.

6.2.2 Property inferencing from video enhanced by 3D skeletons

The small size of unoccluded 3D skeleton motion samples may generate thousands of rendered 2D skeletons. Here we show that these 2D projections of 3D data can effectively improve the performance of property value estimation from 2D video. We generated these virtual 2D samples by projecting the 3D joint positions of 3D skeleton sequences according to different camera viewing angles. For the virtual camera setting, we used a weak-perspective camera model, as suggested by Ref. [58], which generates 2D projections of synthetic 3D skeleton sequences. For every 3D sequence, we used 8 fixed views, placed a camera every 22.5° around the actor (covering 180° in total); all cameras were horizontal (pitch angle 0°).

Table 4 presents the evaluation of 6-class weight classification and 3-class spillability inferencing in the cross-subject setting, using models trained on 2D skeletons extracted from video only, or on 2D extracted skeletons and rendered 3D skeletons. The trained models were tested only on 2D extracted skeletons. In the second case, the ratio of extracted and rendered skeletons was 1:8. Clearly using additional virtual skeletons can effectively improve performance.

Table 4 When adding rendered skeletons to training data, the object inferencing accuracy (%) from video (such as the weight lifted and spillability) is improved significantly

	Lifting (6)	Moving (3)
Without	51.6	62.9
With	**61.4**	**71.4**

6.3 Evaluation of property-aware motion transfer

6.3.1 Setting

We again split the 100 subjects into training (60), validation (20), and test (20) groups, respectively. During training, we selected the network parameters with the smallest validation error over all iterations. We evaluate and report performance on the test groups.

6.3.2 Latent space visualization

Figure 14 shows the latent space of motion samples after projecting the latent features to a 2D image using t-SNE. Each point represents a motion sample of a subject lifting a 0 or 25 kg box. The leftmost figure shows that they are clustered according to object property values without contrastive loss. This is due to the motion differences between different subjects being smaller than those for lifting 0 and 25 kg boxes. With contrastive loss, the features start to disentangle from object properties and become more related to the subjects.

6.3.3 Results

Figures 12, 13, and 15 show three generated motions after changing object property values. Please also

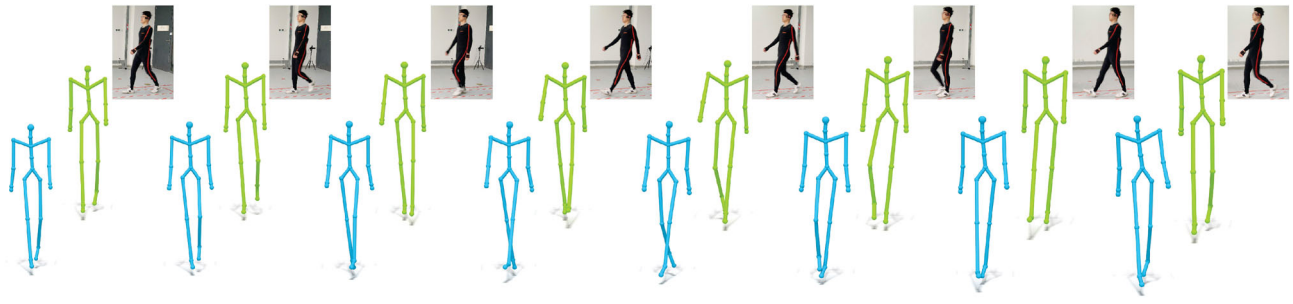


Fig. 12 Given a motion sequence for an unseen subject walking on the wide path (green), we can generate a new sequence that looks like the subject was walking on a narrow path (blue).

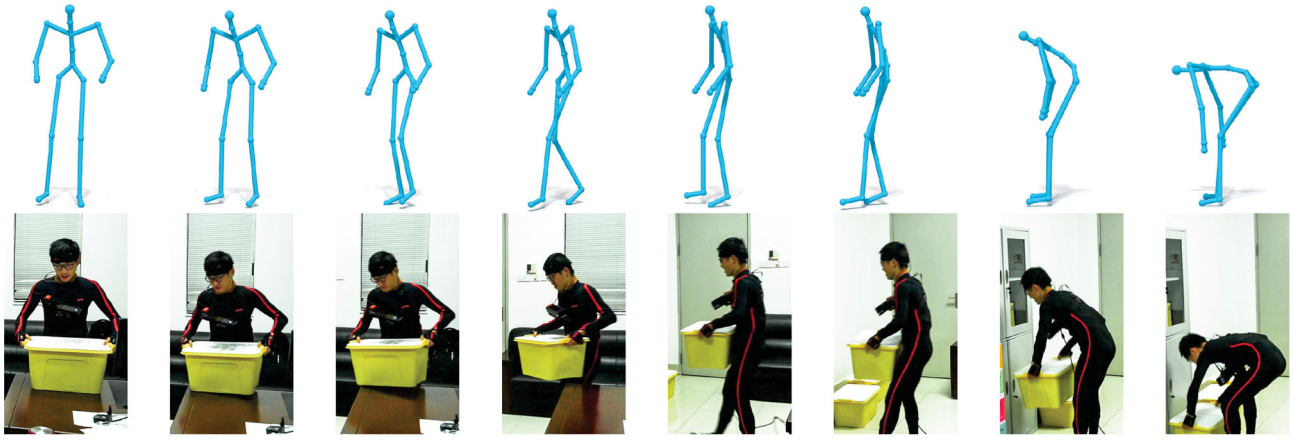


Fig. 13 Given the motion sequence shown in Fig. 1, we can generate a new sequence that looks like the subject was trying to lift a heavy box, but it was too heavy to lift. The generated motion is similar to the ground truth as shown in a sequence of RGB images below.

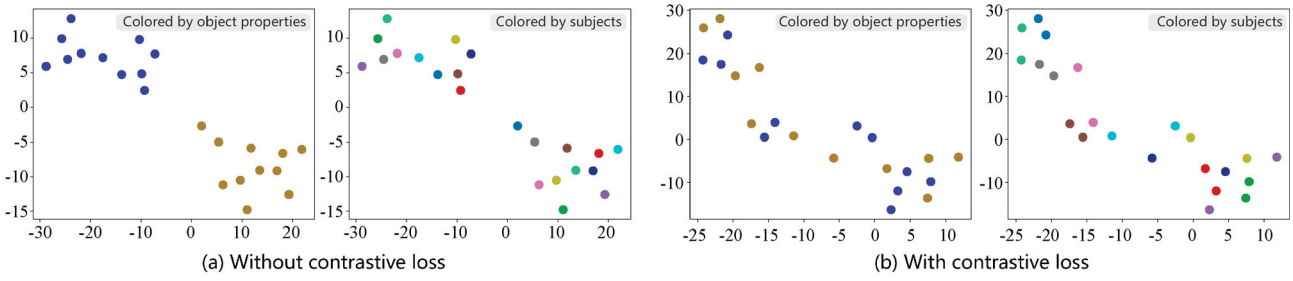


Fig. 14 Latent variables projected to 2D space after encoding of several lifting motions with 0 and 25 kg boxes: (a) without contrastive loss, (b) with contrastive loss, (left) colored by object properties, (right) colored by subject.

refer to the Electronic Supplementary Material for further examples. Given a walking motion on a wide path by an unseen subject, we transfer their motion to walking on a narrow path, like a catwalk model. Given a motion sequence of an unseen subject lifting a light box from a table to a cupboard, we generate a new sequence that looks like the box is too heavy to lift; see Fig. 13.

In Fig. 15, we show a generated sequence of drinking from an empty cup, given an unseen motion sequence of using two hands to drink from a cup full of water. As the unseen motion differs considerably from the training set, the generated motion deviates from the input. However, sometimes the correct motion can be ambiguous. Note that during training, we constrain the synthesized motion conditioned on a target property value to be similar to the motion performed by the same subject for a given object property. Multiple options may likely match the desired motion property value. It would be desirable if we could synthesize the one most similar to the input motion.

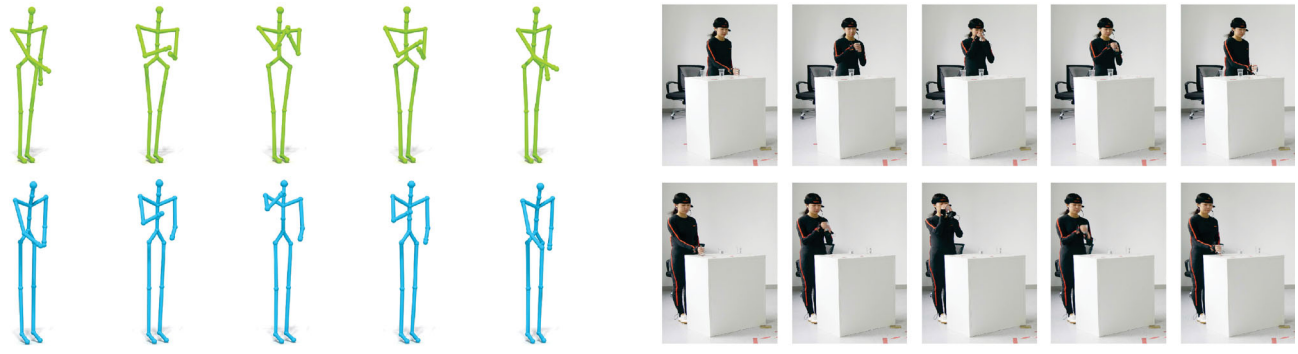


Fig. 15 Given an unseen motion sequence of drinking from a cup full of water using two hands, we generate a new sequence of drinking from an almost empty cup (blue skeletons in the second row). In the training set, all subjects drink water using one hand. Right: corresponding RGB images of the actor for clarity.

6.4 User study

We conducted two user studies. The first investigated a human observer's perception of the weight and spillability inferencing from skeleton sequences. We considered both the 3D skeletons captured and 2D skeletons extracted from video. The second user study evaluated property-aware motion transfer on sitting and walking sequences.

6.4.1 First user study

In this study, the test consisted of watching a video of skeletal motion of an actor lifting a box or moving a bowl, then predicting the unseen object's property by choosing an answer from multiple choices. For Lifting a box sequence, six choices were provided: 0, 5, 10, 15, 20, and 25 kg. For Moving a bowl sequence, three choices were provided: empty, fully filled with rice, and fully filled with water. There were a total of 12 tests. To help answer the questions, 4 samples with correct answers were shown before the tests started. These motion samples were randomly chosen from the testing group. Each video was about 3–6 s long. All participants had full control over these videos: start, pause, stop, navigate in time, etc. A total of 60 participants were recruited. Each participant undertook the user study twice. The first time, they predicted the weight from videos of rendered 3D skeletons, and the second time, they predicted the weight from 2D video skeletons. Note that the 2D video skeletons have large missing parts in certain frames due to occlusion by the human body or the objects themselves, while the rendered ones have much less occlusion, caused by bones. These skeletons were drawn with the same color encoding. The total study time for each participant was around 10 minutes.

Figure 11(left) shows the average weights predicted by users and by our model, for boxes of different physical weights. The weights estimated by our model using 3D skeletons as input were much closer to the physical ground truth than other values. Note that our reported human performance is slightly lower than that reported in Runeson and Frykholm's work [2]. A possible reason is that a smaller weight step (5 kg) and more weight classes (6) were used in our user study. Figure 11 (right) displays the F1 scores of user study and our model on spillability inferencing. It is challenging to distinguish an empty bowl from a bowl full of rice, but still, our model outperformed on both 2D and 3D skeletal cases.

6.4.2 Second user study

A total of 60 participants were recruited and divided into two groups, watching the sitting and walking sequences, respectively. Every participant undertook 12 tests, and 4 samples with correct answers were played before the tests started. A test contained two parts. The first task was to judge if the given motion was synthetic or captured. The second task was to select the associated object property of the given motion, while only 2 choices were provided: for example, to select whether the path being walked on was wide or narrow, or the chair being sat on was soft or hard. Other settings were similar to the first user study. Figure 16 shows the performance of participants on motion source and object property inferencing. The lightness of a square encodes the number of participants with a particular

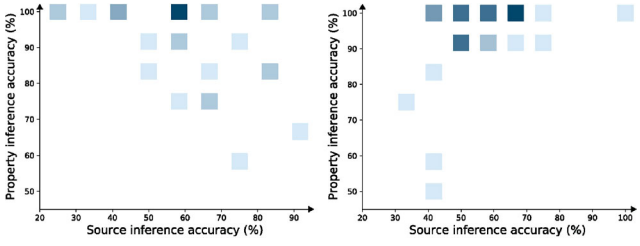


Fig. 16 Scatter map of participants' accuracy (%) on guessing the motion's source (synthesized or captured), and on object property inferencing. Left: sitting; right: walking. Shades of blue indicate the number of participants, darker being higher.

inference accuracy, the darker the higher. For most participants, source inferencing accuracy was about 60%, while property inferencing accuracy was above 90%, indicating that our synthesized motions are quite close to real captured ones.

7 Conclusions and future work

7.1 Summary

The primary goal of this work was to study human interaction motions represented by skeleton sequences. The aims were to investigate whether and how well a machine can learn to infer the properties of unseen interacting objects, and to what extent we can control synthesis of motions with target object properties. We have built a large multi-modal dataset for such object property inferencing from fine-grained human interaction motions with 4000+ samples, consisting of 100 participants performing 8 different tasks, related to 8 different object properties.

Using 3D skeleton sequences alone, we have learned to infer the properties of interacting objects by treating it as a classification problem, and have evaluated our trained model in various settings. The collected 3D skeleton sequences allow data-driven learning, and help achieve better inferencing accuracy compared to using other data sources or even human observers. We have presented a network to disentangle object properties from motion. The disentangling, in turn, allows the synthesis of modified motions with a target object property. This control over the actions enriches the dataset on one hand, and permits the specific animation of particular individuals on the other.

7.2 Limitations

Due to the design, our target problem is limited to defined scenarios with pre-defined human motions and object properties. Inferencing and transfer tasks are solved separately, while exploiting features extracted during inferencing to guide synthesis might work better. The main techniques used in both inferencing and transfer tasks are well established.

Separate classifiers have to be trained for different types of motions, and accuracy is mediocre. We focus only on the intra-class characteristics for object property inferencing, but it might be better to address action recognition and object property inferencing together, as action types provide further global content information.

Our object property-aware motion transfer employs an encoder–decoder structure with 1D convolution layers, which might not fully capture the spatial-temporal information of other human motions, in particular, complex ones. More advanced network structures, such as STRNN [62], could be used to better transfer in-between independent actions.

7.3 Future work

Exciting research directions lay ahead as we are only starting to exploit the collected motion data. We have made a large-scale interaction dataset public. We believe that this dataset will stimulate further research, and in future, we will strive not only to increase the number of samples, but also the types of human–object interaction. Previous works have shown that some other properties, e.g., size and geometric shape, are quite hard to estimate from a pantomimed action [4]. To be able to deal with more diverse object properties, we are also considering fusing other visual inputs, e.g., video and depth sequences, with 3D skeletal motions.

Another promising direction is to discover exactly which parts of the skeleton are critical for each specific object property, by considering more sophisticated attention models or computing more advanced skeletal features. Further exploration could also focus on designing new networks that can learn and encode skeletal motions in a learned latent space, instead of explicitly providing a parameterization. It is certainly more exciting if we can directly predict object properties from 2D video input with large occlusions with high accuracy using a model trained on 3D skeletal motions, eventually leading to new modes of authoring video sequences.

Appendix Interaction motion dataset collection

Walking. The experiment on Walking aims to estimate *the width* of the path. Each subject was asked to walk back and forth on three straight paths of different widths. We simulated the width of a path using line markers to indicate path borders, and asked the subjects to not cross the borders. We have a total of $3 \times 2 \times 100 = 600$ motion samples.

Fishing. The experiment on Fishing aims to estimate *the length* of a fishing rod. Each subject was asked to use a fishing rod to "catch" a magnetic object placed in front of them. The object would attach to the rod's end when touched. Each subject performed 3 tasks, with rods of three different lengths. We have a total of $3 \times 3 \times 100 = 900$ motion samples.

Pouring. The experiment on Pouring aims to estimate *the type* of liquid. Each subject was asked to pour liquid from one cup to another. Each subject did 3 tasks with three different substances (water, shampoo, and rice). The pouring motions were affected by the viscosity or particle granularity.

Bending. The experiment on Bending aims to estimate *the stiffness* of a power twister. Each subject was asked to bend a power twister with three different settings, from easy to hard.

Sitting. The experiment on Sitting aims to estimate *the softness* of a chair. Each subject was asked to sit on four chairs of the same height but different softness. The hardest chair is made of plastic, and the softest one is a yoga ball.

Drinking. The experiment on drinking aims to estimate *the amount* of water inside a cup. Each subject was asked to take a cup from a table and take a sip of water. Each subject performed 3 tasks while the amount of water in the cup changed from almost full, to half full, and to almost empty.

Lifting a box. The experiment on Lifting a box aims to estimate *the weight* of an object from human interaction. Each subject was asked to sequentially perform four different tasks: (i) lifting a box from the ground to a sofa, (ii) lifting the box from the sofa to a table, (iii) lifting the box from the table to the top of a cupboard, and finally (iv) putting the box back on the floor. Without telling the subject, the weight of the box was randomly changed by concealing different weights in the box, ranging from 0 to 25 kg in steps of 5 kg. Each subject performed 6 tasks and did not know if he/she would lift a heavy or light box before each trial, so all captured motions are naturally close to what happens in real life. This experiment provides 1343 motion samples in total, all annotated with the specific task and weight. When a subject failed to lift a heavy box to somewhere high, he/she did not need to perform the following tasks with the same weight.

Moving a bowl. The experiment on Moving a bowl aims to judge the *spillability* of an object from human interactions. While the weight is a physical property, spillability is a more empirical property. Each subject was asked to perform the same four tasks in a row as described above, but to move a bowl rather than lifting a box. Three uncovered bowls of the same kind were used: one empty, one filled with rice, and one filled with water. Thus, each subject performed 3 tasks and could see the different states of these three bowls. They were all required to try their best to move the bowls without spillage. We expect this to capture how cautious the subject was, and how much that correlated to his/her motion in the corresponding trial. The degree of caution should be highest when moving a bowl full of water, and the lowest when moving an empty bowl, corresponding to the spillability. All action samples were annotated with one of the three levels of spillability.

Acknowledgements

We sincerely thank the reviewers for their valuable comments. This work was supported in part by Shenzhen Innovation Program (JCYJ20180305125709986), National Natural Science Foundation of China (61861130365, 61761146002), GD Science and Technology Program (2020A0505100064, 2015A030312015), and DEGP Key Project (2018KZDXM058).

Electronic Supplementary Material Supplementary material is available in the online version of this article at `https://doi.org/10.1007/s41095-021-0218-8`.

References

[1] Blake, R.; Shiffrar, M. Perception of human motion. *Annual Review of Psychology* Vol. 58, No. 1, 47–73, 2007.

[2] Runeson, S.; Frykholm, G. Visual perception of lifted weight. *Journal of Experimental Psychology: Human*

Perception and Performance Vol. 7, No. 4, 733–740, 1981.

[3] Podda, J.; Ansuini, C.; Vastano, R.; Cavallo, A.; Becchio, C. The heaviness of invisible objects: Predictive weight judgments from observed real and pantomimed grasps. *Cognition* Vol. 168, 140–145, 2017.

[4] Vaina, L. M.; Goodglass, H.; Daltroy, L. Inference of object use from pantomimed actions by aphasics and patients with right hemisphere lesions. *Synthese* Vol. 104, No. 1, 43–57, 1995.

[5] Shahroudy, A.; Liu, J.; Ng, T. T.; Wang, G. NTU RGB+D: A large scale dataset for 3D human activity analysis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1010–1019, 2016.

[6] Liu, C.; Hu, Y.; Li, Y.; Song, S.; Liu, J. PKU-MMD: A large scale benchmark for continuous multi-modal human action understanding. *arXiv preprint* arXiv:1703.07475, 2017.

[7] Lo Presti, L.; La Cascia, M. 3D skeleton-based human action classification: A survey. *Pattern Recognition* Vol. 53, 130–147, 2016.

[8] Yao, B. P.; Fei-Fei, L. Modeling mutual context of object and human pose in human-object interaction activities. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 17–24, 2010.

[9] Gkioxari, G.; Girshick, R.; Dollár, P.; He, K. Detecting and recognizing human-object interactions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8359–8367, 2018.

[10] Kato, K.; Li, Y.; Gupta, A. Compositional learning for human object interaction. In: *Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11218*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 247–264, 2018.

[11] Grabner, H.; Gall, J.; van Gool, L. What makes a chair a chair? In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1529–1536, 2011.

[12] Kim, V. G.; Chaudhuri, S.; Guibas, L.; Funkhouser, T. Shape2Pose. *ACM Transactions on Graphics* Vol. 33, No. 4, Article No. 120, 2014.

[13] Hu, R. Z.; Yan, Z. H.; Zhang, J. W.; van Kaick, O., Shamir, A., Zhang, H.; Huang, H. Predictive and generative neural networks for object functionality. *ACM Transactions on Graphics* Vol. 37, No. 4, Article No. 151, 2018.

[14] Savva, M.; Chang, A. X.; Hanrahan, P.; Fisher, M.; Nießner, M. SceneGrok. *ACM Transactions on Graphics* Vol. 33, No. 6, Article No. 212, 2014.

[15] Li, X. T.; Liu, S. F.; Kim, K.; Wang, X. L.; Yang, M. H.; Kautz, J. Putting humans in a scene: Learning affordance in 3D indoor environments. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 12360–12368, 2019.

[16] Hu, R.; Savva, M.; van Kaick, O. Functionality representations and applications for shape analysis. *Computer Graphics Forum* Vol. 37, No. 2, 603–624, 2018.

[17] Jiang, Y.; Koppula, H.; Saxena, A. Hallucinated humans as the hidden context for labeling 3D scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2993–3000, 2013.

[18] Jiang, Y.; Koppula, H. S.; Saxena, A. Modeling 3D environments through hidden human context. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 38, No. 10, 2040–2053, 2016.

[19] Ho, E. S. L.; Komura, T.; Tai, C. L. Spatial relationship preserving character motion adaptation. *ACM Transactions on Graphics* Vol. 29, No. 4, Article No. 33, 2010.

[20] Shen, Y. J.; Yang, L. Z.; Ho, E. S. L.; Shum, H. P. H. Interaction-based human activity comparison. *IEEE Transactions on Visualization and Computer Graphics* Vol. 26, No. 8, 2620–2633, 2019.

[21] Stoffregen, T. A.; Flynn, S. B. Visual perception of support-surface deformability from human body kinematics. *Ecological Psychology* Vol. 6, No. 1, 33–64, 1994.

[22] Hamilton, A. F.; Joyce, D. W.; Flanagan, J. R.; Frith, C. D.; Wolpert, D. M. Kinematic cues in perceptual weight judgement and their origins in box lifting. *Psychological Research* Vol. 71, No. 1, 13–21, 2007.

[23] Schmidt, F.; Paulun, V. C.; van Assen, J. J. R.; Fleming, R. W. Inferring the stiffness of unfamiliar objects from optical, shape, and motion cues. *Journal of Vision* Vol. 17, No. 3, 18, 2017.

[24] Koppula, H. S.; Gupta, R.; Saxena, A. Learning human activities and object affordances from RGB-D videos. *The International Journal of Robotics Research* Vol. 32, No. 8, 951–970, 2013.

[25] Kang, C. G.; Lee, S. H. Scene reconstruction and analysis from motion. *Graphical Models* Vol. 94, 25–37, 2017.

[26] Monszpart, A.; Guerrero, P.; Ceylan, D.; Yumer, E.; Mitra, N. J. iMapper: Interaction-guided joint scene and human motion mapping from monocular videos. *ACM Transactions on Graphics* Vol. 38, No. 4, Article No. 92, 2019.

[27] Davis, J. W.; Gao, H. Recognizing human action efforts: An adaptive three-mode PCA framework. In: Proceedings of the 9th IEEE International Conference on Computer Vision, 1463–1469, 2003.

[28] Gupta, A.; Davis, L. S. Objects in action: An approach for combining action understanding and object perception. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1–8, 2007.

[29] Wu, J.; Yildirim, I.; Lim, J. J.; Freeman, W. T.; Tenenbaum, J. B. Galileo: Perceiving physical object properties by integrating a physics engine with deep learning. In: Proceedings of the 28th International Conference on Neural Information Processing Systems, Vol. 1, 127–135, 2015.

[30] Wu, J. J.; Lim, J.; Zhang, H. Y.; Tenenbaum, J.; Freeman, W. Physics 101: Learning physical object properties from unlabeled videos. In: Proceedings of the British Machine Vision Conference, 39.1–39.12, 2016.

[31] Liu, J.; Shahroudy, A.; Xu, D.; Wang, G. Spatio-temporal LSTM with trust gates for 3D human action recognition. In: *Computer Vision – ECCV 2016. Lecture Notes in Computer Science, Vol. 9907*. Leibe, B.; Matas, J.; Sebe, N.; Welling, M. Eds. Springer Cham, 816–833, 2016.

[32] Liu, J.; Wang, G.; Hu, P.; Duan, L. Y.; Kot, A. C. Global context-aware attention LSTM networks for 3D action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3671–3680, 2017.

[33] Song, S.; Lan, C.; Xing, J.; Zeng, W.; Liu, J. An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In: Proceedings of the 31st AAAI Conference on Artificial Intelligence, 4263–4270, 2017.

[34] Yan, S. J.; Xiong, Y. J.; Lin, D. H. Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2018.

[35] Ke, Q. H.; Bennamoun, M.; An, S. J.; Sohel, F.; Boussaid, F. A new representation of skeleton sequences for 3D action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4570–4579, 2017.

[36] Li, C.; Zhong, Q. Y.; Xie, D.; Pu, S. L. Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence, 786–792, 2018.

[37] Aristidou, A.; Cohen-Or, D.; Hodgins, J. K.; Chrysanthou, Y.; Shamir, A. Deep motifs and motion signatures. *ACM Transactions on Graphics* Vol. 37, No. 6, Article No. 187, 2018.

[38] Hsu, E.; Pulli, K.; Popović, J. Style translation for human motion. *ACM Transactions on Graphics* Vol. 24, No. 3, 1082–1089, 2005.

[39] Xia, S. H.; Wang, C. Y.; Chai, J. X.; Hodgins, J. Realtime style transfer for unlabeled heterogeneous human motion. *ACM Transactions on Graphics* Vol. 34, No. 4, Article No. 119, 2015.

[40] Yumer, M. E.; Mitra, N. J. Spectral style transfer for human motion between independent actions. *ACM Transactions on Graphics* Vol. 35, No. 4, Article No. 137, 2016.

[41] Bellini, R.; Kleiman, Y.; Cohen-Or, D. Dance to the beat: Synchronizing motion to audio. *Computational Visual Media* Vol. 4, No. 3, 197–208, 2018.

[42] Cao, Z.; Simon, T.; Wei, S. H.; Sheikh, Y. Realtime multi-person 2D pose estimation using part affinity fields. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1302–1310, 2017.

[43] Insafutdinov, E.; Pishchulin, L.; Andres, B.; Andriluka, M.; Schiele, B. DeeperCut: A deeper, stronger, and faster multi-person pose estimation model. In: *Computer Vision – ECCV 2016. Lecture Notes in Computer Science, Vol. 9910*. Leibe, B.; Matas, J.; Sebe, N.; Welling, M. Eds. Springer Cham, 34–50, 2016.

[44] Newell, A.; Yang, K.; Deng, J. Stacked hourglass networks for human pose estimation. In: *Computer Vision – ECCV 2016. Lecture Notes in Computer Science, Vol. 9912*. Leibe, B.; Matas, J.; Sebe, N.; Welling, M. Eds. Springer Cham, 483–499, 2016.

[45] Wei, S. H.; Ramakrishna, V.; Kanade, T.; Sheikh, Y. Convolutional pose machines. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4724–4732, 2016.

[46] Güler, R. A.; Neverova, N.; Kokkinos, I. DensePose: Dense human pose estimation in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7297–7306, 2018.

[47] Tekin, B.; Rozantsev, A.; Lepetit, V.; Fua, P. Direct prediction of 3D body poses from motion compensated sequences. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 991–1000, 2016.

[48] Tome, D.; Russell, C.; Agapito, L. Lifting from the deep: Convolutional 3D pose estimation from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5689–5698, 2017.

[49] Mehta, D.; Sridhar, S.; Sotnychenko, O.; Rhodin, H.; Shafiei, M.; Seidel, H. P.; Xu, W.; Casas, D.; Theobalt, C. VNect: Real-time 3D human pose estimation with a single RGB camera. *ACM Transactions on Graphics* Vol. 36, No. 4, Article No. 44, 2017.

[50] Kanazawa, A.; Black, M. J.; Jacobs, D. W.; Malik, J. End-to-end recovery of human shape and pose. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7122–7131, 2018.

[51] Pavlakos, G.; Zhou, X. W.; Daniilidis, K. Ordinal depth supervision for 3D human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7307–7316, 2018.

[52] Andriluka, M.; Iqbal, U.; Insafutdinov, E.; Pishchulin, L.; Milan, A.; Gall, J.; Schiele, B. PoseTrack: A benchmark for human pose estimation and tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5167–5176, 2018.

[53] CMU. CMU Graphics Lab Motion Capture Database. 2018. Available at http://mocap.cs.cmu.edu/.

[54] Cho, K.; van Merrienboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 1724–1734, 2014.

[55] Wang, Y.; Sun, Y. B.; Liu, Z. W.; Sarma, S. E.; Bronstein, M. M.; Solomon, J. M. Dynamic graph CNN for learning on point clouds. *ACM Transactions on Graphics* Vol. 38, No. 5, Article No. 146, 2019.

[56] Zhang, P. F.; Xue, J. R.; Lan, C. L.; Zeng, W. J.; Gao, Z. N.; Zheng, N. N. Adding attentiveness to the neurons in recurrent neural networks. In: *Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11213*. Springer Cham, 136–152, 2018.

[57] Holden, D.; Saito, J.; Komura, T. A deep learning framework for character motion synthesis and editing. *ACM Transactions on Graphics* Vol. 35, No. 4, Article No. 138, 2016.

[58] Aberman, K.; Wu, R. D.; Lischinski, D.; Chen, B. Q.; Cohen-Or, D. Learning character-agnostic motion for motion retargeting in 2D. *ACM Transactions on Graphics* Vol. 38, No. 4, Article No. 75, 2019.

[59] Gui, L. Y.; Wang, Y. X.; Liang, X. D.; Moura, J. M. F. Adversarial geometry-aware human motion prediction. In: *Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11208*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 823–842, 2018.

[60] Hadsell, R.; Chopra, S.; LeCun, Y. Dimensionality reduction by learning an invariant mapping. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1735–1742, 2006.

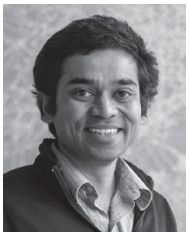
[61] Charles, R. Q.; Hao, S.; Mo, K. C.; Guibas, L. J. PointNet: Deep learning on point sets for 3D classification and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 77–85, 2017.

[62] Wang, H.; Ho, E. S. L.; Shum, H. P. H.; Zhu, Z. X. Spatio-temporal manifold learning for human motions via long-horizon modeling. *IEEE Transactions on Visualization and Computer Graphics* Vol. 27, No. 1, 216–227, 2021.

Qian Zheng received her doctoral degree in computer science from Shenzhen Institutes of Advanced Technology (SIAT), Chinese Academy of Sciences, in 2015. She is an assistant professor in the College of Computer Science and Software Engineering, Shenzhen University. Her interests include computer graphics and information visualization.

Weikai Wu is a software engineer in TCL. He received his M.S. degree in computer science from Shenzhen University in 2020.

Hanting Pan is a software engineer in Orbbec. He received his M.S. degree in computer science from Shenzhen University in 2020.



Niloy Mitra leads the Smart Geometry Processing Group in the Department of Computer Science at University College London. He received his Ph.D. degree from Stanford University under the guidance of Leonidas Guibas. His research interests include shape analysis, creativeAI, and computational design and fabrication. Niloy received the Eurographics Outstanding Technical Contributions Award in 2019, the

BCS Roger Needham Award in 2015, and the ACM Siggraph Significant New Researcher Award in 2013.

Daniel Cohen-Or is a professor in the School of Computer Science, Tel Aviv University. He received his Ph.D. degree from the State University of New York at Stony Brook in 1991. He was the recipient of a Eurographics Outstanding Technical Contributions Award in 2005, and an ACM SIGGRAPH Computer Graphics Achievement Award in 2018. In 2019 he won a Kadar Family Award for Outstanding Research. In 2020 he received a Eurographics Distinguished Career Award. His research interests are in computer graphics, in particular, synthesis, processing, and modeling techniques.

Hui Huang is a Distinguished TFA Professor at Shenzhen University, where she directs the Visual Computing Research Center. She received her Ph.D. degree in applied math from The University of British Columbia in 2008. Her research interests span computer graphics, 3D vision, and visualization. She is currently a Senior Member of IEEE/ACM/CSIG, a Distinguished Member of CCF, and is on the editorial board of *ACM Trans. on Graphics* and *Computers & Graphics*.