Task-Oriented Human-Object Interactions Generation with Implicit Neural Representations

∗ *Supplementary Material* ∗

Quanzhou Li[1] Jingbo Wang[2] Chen Change Loy[1] Bo Dai[2]
[1] S-Lab, Nanyang Technological University [2] Shanghai AI Laboratory
{quanzhou001, ccloy}@ntu.edu.sg {wangjingbo, daibo}@pjlab.org.cn

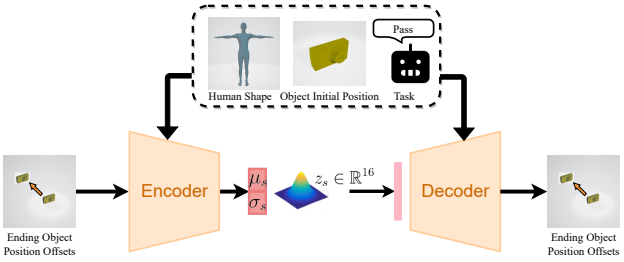


Figure 1. Overview of our object parameters sampler. The encoder encodes the inputs to a normal distribution parameterized by μ_s and σ_s. The decoder then samples a latent code from the distribution and uses it with the conditioned information to estimate the ending object position offsets.

Our supplementary material includes this document and a video. Since our work synthesizes motions that involve 3D humans and objects in simulation, it is best to evaluate the realism of the generated sequences through dynamic presentation. Our video includes a brief introduction to the project's motivation, the problem formulation, the overview of our method, and qualitative results.

1. Data Preparation

We train our pipeline on the GRAB dataset [6] similar to previous works [5, 7]. GRAB contains full 3D human shape and pose sequences of 10 digital humans interacting with 51 objects of various shapes and sizes with four intents, namely pass, lift, offhand, and use. Following [5], we take out 4 validation objects (*fryingpan, toothbrush, elephant, hand*) and 5 test objects (*mug, camera, binoculars, apple, toothpaste*). To calculate the object BPS representation, we center the object point cloud at its geometry center and use 1024 vertices sampled from $[-0.15, 0.15]^3$ as a fixed basis point set for all objects.

We manually label the keyframes of each task. The core idea of labeling is to select the frames of stable right-hand grasps that perceptually well represent the task. For each sequence in the dataset, we select frames within a threshold set empirically for each task and label them as the corresponding task keyframes.

As the motion of approaching objects from T-poses and the motion of conducting tasks from grasping usually take about 2 seconds in the GRAB dataset, we downsample the framerate of GRAB from 120 to 30 and clip 64-frame motion clips to train our motion inbetweening network. We use four markers on each foot to compute foot-ground contact labels $C_{fg} \in \{0, 1\}^8$. The marker is considered in contact with the ground when it is within 5cm of the ground and its velocity is less than 75mm/s. To evaluate our motion inbetweening network, we additionally train and test it on the AMASS dataset [3] following [7] for fair comparisons. AMASS is a large-scale human motion dataset that captures more than 11000 motions.

2. Network Architectures

2.1. Object Parameters Sampler

Our object parameters sampler is a cVAE conditioned on the task type, the human shape, and the object's initial translation and orientation. Figure 1 shows the architecture of the sampler. The encoder encodes the conditioned information with the ground truth translation and orientation offsets to a latent space of 16 dimensionalities. The decoder then uses the conditioned information and a sampled code from the latent space to reconstruct the translation and orientation offsets. Both the encoder and decoder are implemented using fully-connected layers with skip connections.

2.2. Motion Inbetweening Network

Figure 2 shows the architecture of the motion inbetweening network. Our model takes the poses of the first and last frames and their translation distance as inputs and generates an infilled motion. We train the motion inbetweening net with 64-frame data, but in inference, the model can synthesize motions of arbitrary frames. For training, we lin-

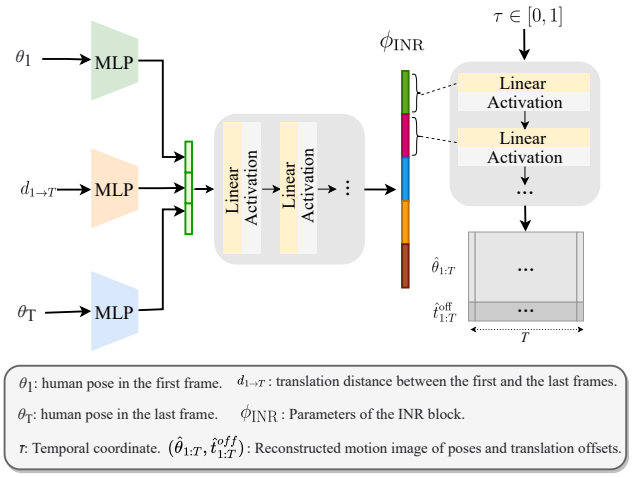


Figure 2. Overview of our motion inbetweening network. The model is bypernetwork-based. It takes the human poses in the first and last frames, θ_1 and θ_T, and their translation distance $d_{1\to T}$ as inputs and generates the weights of an INR block. The INR then takes the temporal coordinates from $[0, 1]$ to evaluate the translation offsets from the interpolated trajectory and the human poses at corresponding timestamps.

early interpolate the trajectory using the translations of the first and last frames and only predicts the offsets from it. In inference, the interpolation scheme depends on the input temporal coordinate vector τ, e.g., if upsamples to two times the original frames, the interpolation scheme will output the corresponding interpolated trajectory linearly two times denser. We use the same marker placement as [7] to select the surface body markers.

The weights prediction module of the motion inbetweening net is implemented by 10 skip-connected linear layers with hidden dimensions of 2048, accompanied by a connector linear layer to output full parameters used by the INR as a vector, which is then truncated in parts as weights of each of the INR layers. The pose parameters θ_1 and θ_T passed to the weights prediction module are flattened to vectors in $\mathbf{R}^{330}$ as the first and last columns of the motion image. We apply factorized multiplicative modulation [4] and the squeeze-and-excitation mechanism [2] to implement the INR, which reduces the number of parameters of the INR and the computation costs for the connector layer to predict the weights.

3. Experiments

3.1. Qualitative Results

We show additional generated results in Figure 3, 4, 5, and 6, and present more complete generated sequences in our supplemental video. We also show the upsampling and velocity adjustment results in the video. For upsampling, we uniformly divide the temporal coordinate interval $[0, 1]$ into segments equal to the number of frames. For the non-uniform velocity adjustment, we divide the temporal coordinate interval into T sub-intervals, where T is the number of frames, such that the lengths of the sub-intervals form a geometric sequence, i.e., the lengths of the sub-intervals satisfy: $l_n = l_1 \cdot r^{n-1}$, where l_n is the length of the n-th sub-interval, $n \in \{1, ..., T\}$, r is the ratio of the geometric sequence. For the slow-to-fast sequence, we define $r = 1.09$. For the fast-to-slow sequence, we define $r = 1/1.09$

3.2. Quantitative Evaluation Metrics

APD. We use the Average L2 Pairwise Distance (APD) [8] to measure the diversity within generated samples. The APD is computed by:

$$\frac{1}{K(K-1)} \sum_{i=1}^{K} \sum_{j \neq i}^{K} ||x_i - x_j||_2, \tag{1}$$

where K is number of samples and x_i, $i \in \{1, ..., K\}$, are the sampled data.

ADE. We compute the Average L2 Distance (ADE) between the reconstructed marker sequences and the ground truth. The formula to compute ADE is:

$$\frac{1}{T} \sum_{i=1}^{T} ||x_i - \hat{x}_i||_2, \tag{2}$$

where T is the number of frames, x_i and $\hat{x}_i$ are the ground truth and reconstructed values respectively.

PSKL-J. Power Spectrum KL Divergence (PSKL) [1] is used to measure the distribution distance between our generated results and the ground truth motion sequences. Here, we follow [9] to evaluate PSKL w.r.t. the acceleration distribution for the SMPL-X joints (PSKL-J). For a frame of F features, the power spectrum of each feature sequence s_f is computed as $\text{PS}(s_f) = ||\text{FFT}(s_f)||^2$. Thus, the average power spectrum of the feature s_f over N motion sequences on one dataset C is given by:

$$\text{PS}(C|f) = \frac{1}{N} \sum_{n=1}^{N} \text{PS}(s_f) \tag{3}$$

and the PSKL between the ground truth and generated datasets is computed by:

$$\text{PSKL}(C, D) = \frac{1}{F} \sum_{f=1}^{F} \sum_{e=1}^{E} ||\text{PS}(C|f)|| * \log\left(\frac{||\text{PS}(C|f)||}{||\text{PS}(D|f)||}\right), \tag{4}$$

where C and D are datasets, f is a feature, and e is frequency. As PSKL is asymmetric, we compute both directions to demonstrate the results.

3.3. Failure Cases and Limitations

Even though TOHO generates continuous and complete human-object manipulation sequences, we observe some failure cases when the goal net outputs inaccurate grasping poses. In Fig. 7, we show an example that the human hand penetrating the object. Besides, although our object motion estimation algorithm calculating object trajectory by keeping the hand-object relationship of the grasping frame works well on the GRAB dataset, the modeling of in-hand object manipulation is limited in our setting. Another limitation of TOHO is that it focuses on the modeling the human and the interacting object while ignoring the environment objects, which sometimes can lead to collisions with the surrounding objects as shown in Fig. 8.

4. Social Impact

Generating realistic human-object manipulation motions is of great value and interest from computer vision to robotics. It has affluent applications in AR/VR, movies, and video games. We recognize that although most application scenarios of such frameworks are positive, harmful exercises of these technologies may lead to destructive behaviors, especially with the advancement of deepfakes and neural rendering. We will make the research available with the appropriate license to prevent the framework from being used by whole-body deepfakes.

References

[1] A. Hernandez, J. Gall, and F. Moreno-Noguer. Human motion prediction via spatiotemporal inpainting. In *International Conference on Computer Vision (ICCV)*, 2019. 2

[2] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[3] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision (ICCV)*, 2019. 1

[4] I. Skorokhodov, S. Ignatyev, and M. Elhoseiny. Adversarial generation of continuous images. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[5] O. Taheri, V. Choutas, M.J. Black, and D. Tzionas. GOAL: Generating 4D whole-body motion for hand-object grasping. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1

[6] O. Taheri, N. Ghorbani, M.J. Black, and D. Tzionas. GRAB: A dataset of whole-body human grasping of objects. In *European Conference on Computer Vision (ECCV)*, 2020. 1

[7] Y. Wu, J. Wang, Y. Zhang, S. Zhang, O. Hilliges, F. Yu, and S. Tang. SAGA: Stochastic whole-body grasping with contact. In *European Conference on Computer Vision (ECCV)*, 2022. 1, 2

[8] Y. Yuan and K. Kitani. Dlow: Diversifying latent flows for diverse human motion prediction. In *European Conference on Computer Vision (ECCV)*, 2020. 2

[9] S. Zhang, Y. Zhang, F. Bogo, M. Pollefeys, and S. Tang. Learning motion priors for 4d human body capture in 3d scenes. In *International Conference on Computer Vision (ICCV)*, 2021. 2

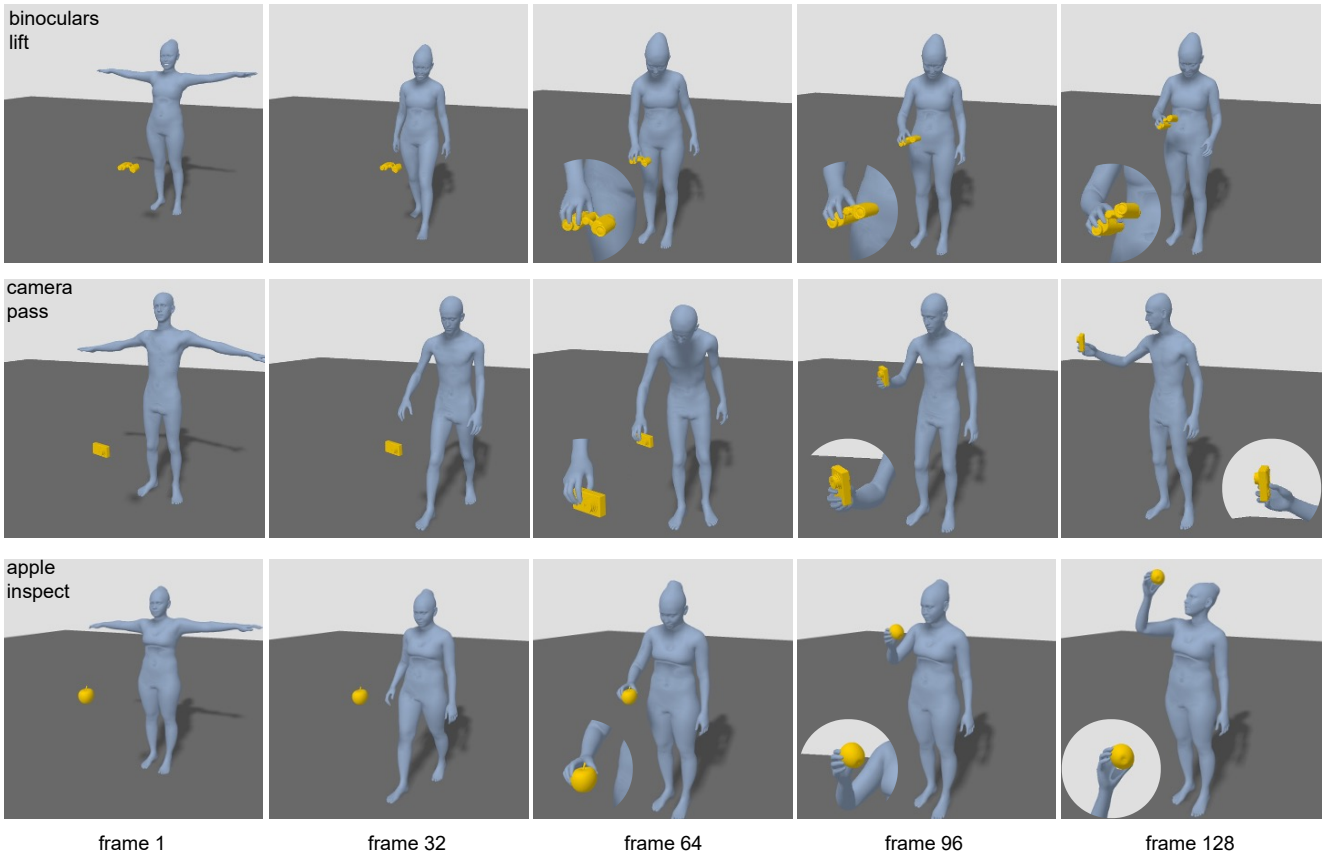


Figure 3. Additional generated results by our framework. We show generated motion sequences of humans of **distinct shapes** conducting **different tasks** with **unseen objects**.

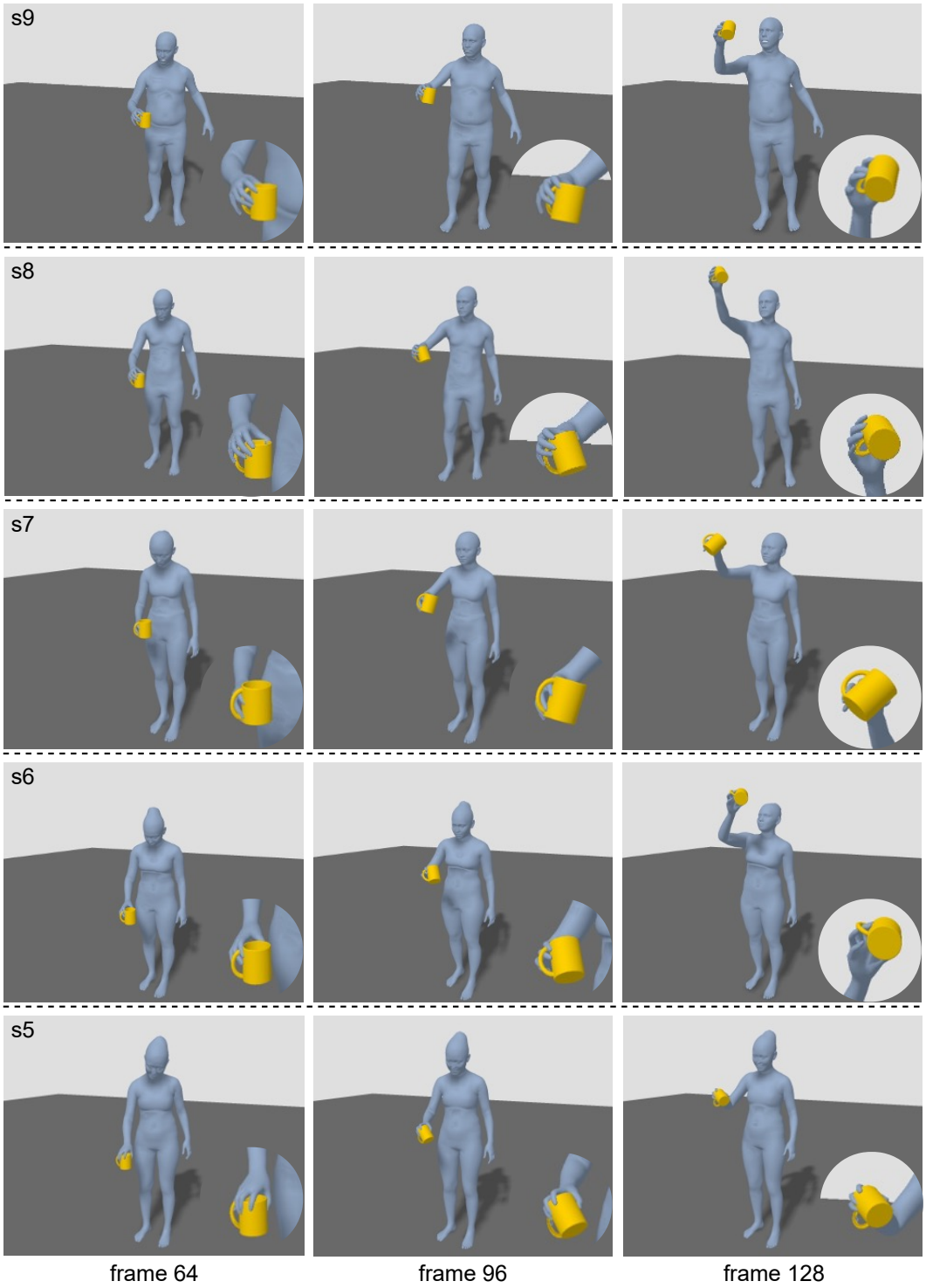


Figure 4. From the top row to the bottom, we display the generated results of humans of **various body types** performing the same task with the same object.

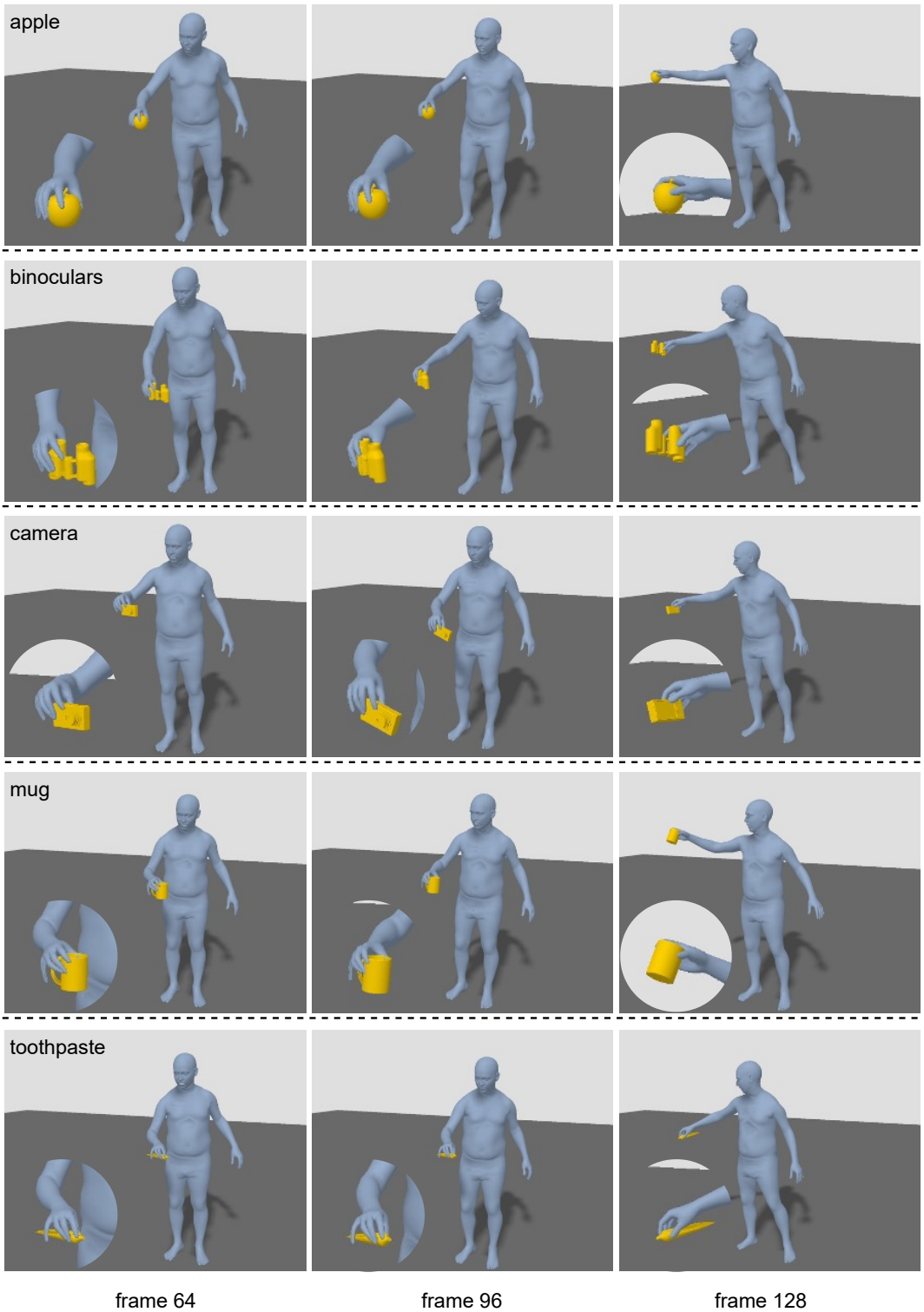
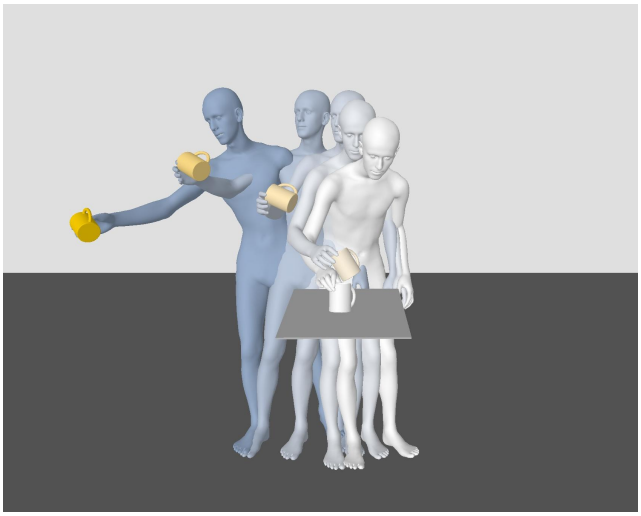
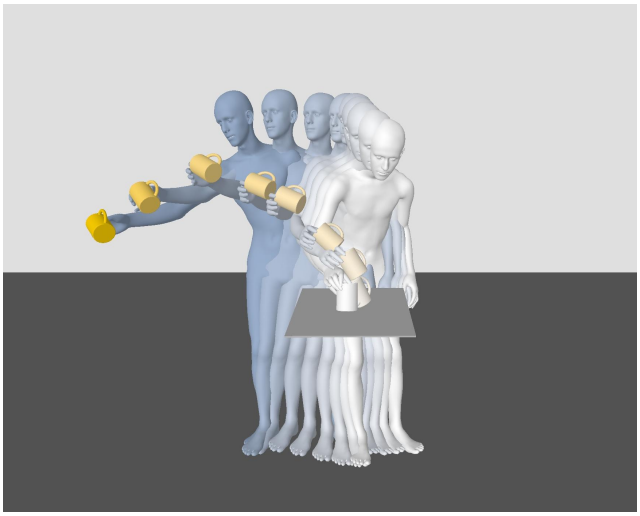


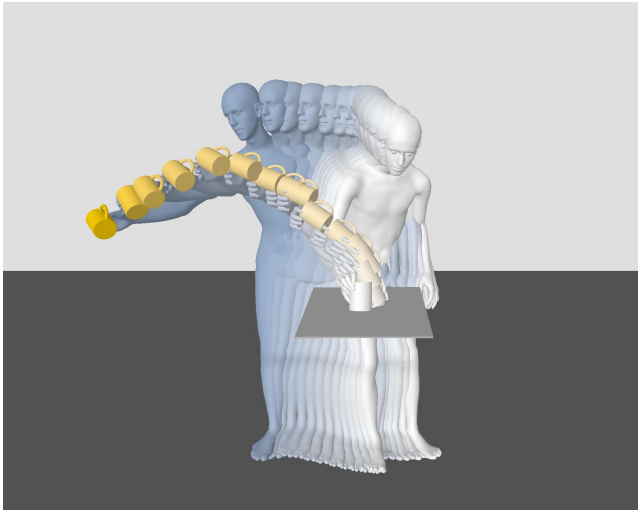
Figure 5. From the top to the bottom row, we show generated results of the same human conducting the same task with **different unseen objects**.



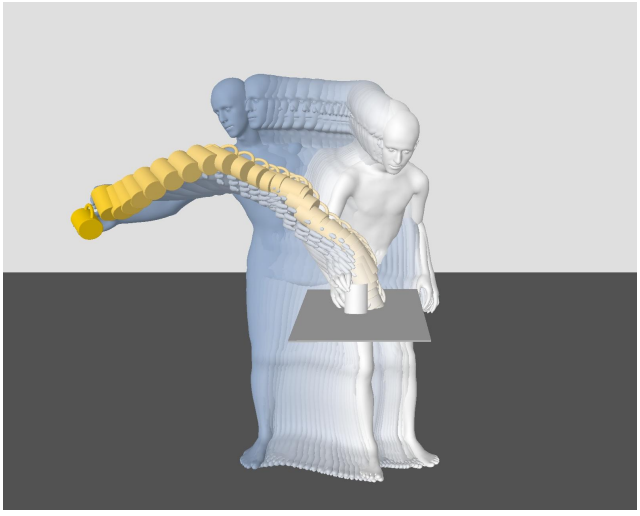
(a) 64 frames.



(b) 128 frames.



(c) 256 frames.



(d) 512 frames.

Figure 6. Examples of upsampling generated motion sequences to higher frames. Fig. (a) to (d) are the results of 64, 128, 256, and 512 frames, respectively. The results demonstrate our method can generate motions of framerates well beyond the training data. We offer the results with a skip frame of 16 and include the last frames.

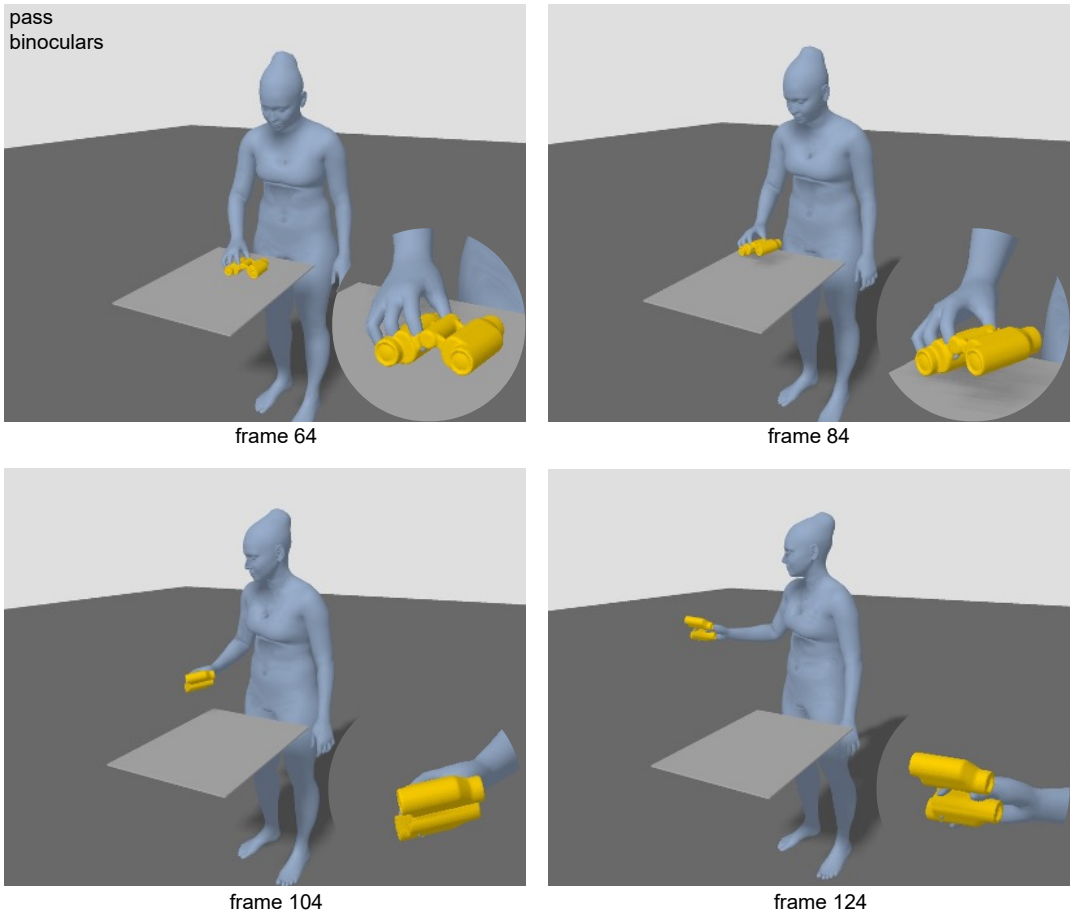


Figure 7. A failure case. As our object motion estimation algorithm computes the object trajectory by keeping the hand-object relationship of the grasping frame across the motion, hand-object penetration will be kept as well when an inaccurate grasping pose is given.

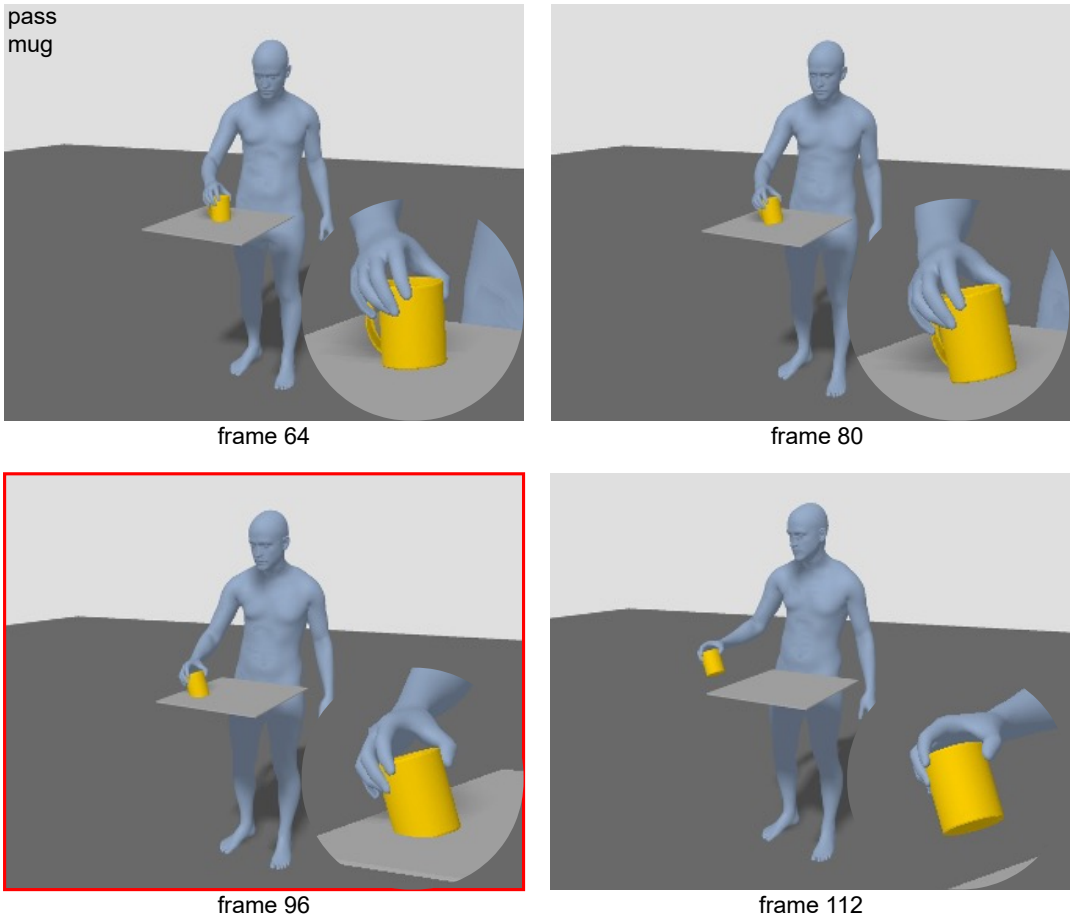


Figure 8. A limitation of our framework. The focus of this work is to generate human-object manipulation motions and the environment is not taken into account, which may sometimes lead to object-table collisions.