

FORCE: Physics-aware Human-object Interaction

Supplementary Material

Paper ID 142

In this document, we provide explanations for the list of symbols (Section 1). Subsequently, we present a failure case of our method (Section 2). Furthermore, we offer additional details on the dataset (Section 3) and the implementation and training process (Section 4). We encourage readers to refer to our supplementary video for animated qualitative results.

1 Comparison with InterDiff

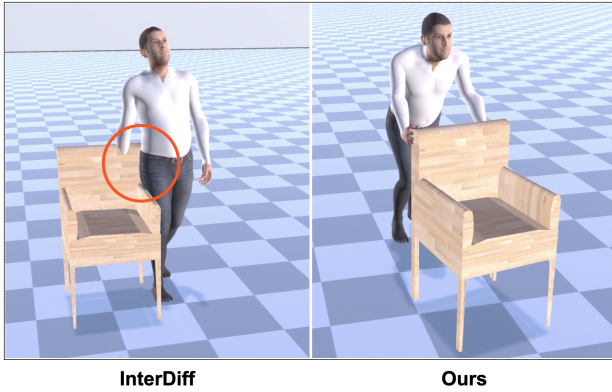


Figure 1. Qualitative comparison with InterDiff [7].

As shown in Figure 1, we compare our method FORCE with InterDiff [7], a state-of-the-art human-object interaction method with diffusion model. We compare under the offline human motion synthesis setup on testing sequences. It can be seen, since InterDiff is *not* designed to handle goal-reaching motion, the interaction may fail.

Computationally, FORCE runs in real-time, significantly ($16\times$) faster than InterDiff, since InterDiff requires iterated denoising as part of the diffusion model.

2 Failure Case

The object augmentation that we employed for training enables FORCE to generalize to unseen object shapes at testing (see all the animations in the supplementary video.) However, when the object shape is too large, there may exist



Figure 2. The interaction exhibits artifact when the object shape is too large.

interpenetration artifact.

3 Dataset

Details on the dataset: Our dataset comprises 450 motion sequences involving human-object interactions with a diverse range of resistance forces. Table 1 provides a detailed breakdown of the dataset categorized by the level of resistance. In this context, resistance is measured solely by the mass of the removable weight used. The masses of the objects themselves are not factored into these measurements. They are measured and will be provided with the dataset. The dataset distributions based on the type of action (Table 2), and the type of hand contact (Table 3) are also presented.

Table 1. Distribution of the dataset by the level of resistance. The data is categorized by the mass of removable weight used. Note, the masses of the objects themselves are not factored into these measurements.

Mass	Minutes	%
0 kg	47.1	33.3
5 kg	18.2	12.9
10 kg	21.4	15.1
15 kg	23.8	16.8
20 kg	7.9	5.6
25 kg	8.9	6.2
>30 kg	14.2	10.1

Details on human tracking. The first stage of our human

Table 2. Distribution of the dataset by the type of action.

Action Type	Minutes	%
Carry	107.3	75.9
Push	17.4	12.3
Pull	16.7	11.8

Table 3. Distribution of the dataset with different hand contact.

Interaction Type	Minutes	%
Right Hand	22.7	19.4
Left Hand	27.4	16.0
Both Hand	91.4	64.6

tracking is to fit the SMPL parametric model [4] to the point clouds captured by the Kinect cameras. We segment humans in captured RGB images using Detectron V2 [6]. The resulting masks are then used to segment the human from the RGB data, before the the human point cloud is lifted in 3D. To initialize the SMPL pose, we employ FrankMocap [5] from the images. Subsequently, instance-specific optimization techniques [1] are applied to fit the SMPL model to the segmented human point cloud via ICP. For more precise fitting, we further derive the SMPL shape parameters of each subject from 3D scans using [2]. This stage produces the SMPL parameters fitted to the cameras, but they can be noisy and erroneous due to occlusion. The second stage of our tracking is to refine the IMU-captured motion, which is smoother and more robust against occlusion. We synchronize the IMU-captured motion with the Kinect-fitted results from the previous stage, then perform an optimization to further refine the IMU-captured motion with the previously fitted results. The resulting motion is smooth and accurately captures the contact between the human and the object.

4 Architecture and Training Details

The motion synthesis network, *MNet*, adopts a mixture-of-expert structure [3]. Both the gating network and the prediction networks consist of three-layer fully-connected networks, with hidden dimensions of 128 and 512, respectively. The model employs 8 experts and is trained for 150 epochs using an Adam optimizer. The initial learning rate is set at $1e-4$, and a cosine learning rate scheduler gradually reduces it to $5e-6$. A batch size of 32 is utilized, and the complete training process takes approximately 9 hours on an NVIDIA V100 GPU.

The contact prediction network, *CNet* encodes the object geometry \mathbf{G} through a three-layer fully connected network of shape $\{512, 512, 64\}$, the resistance \mathbf{R} , human joint positions \mathbf{j}_i^p and desired action \mathbf{o}_i^a in a separate network with identical shape. The latent vector \mathbf{z} of the VAE is of size 6. The weight of the Kullback-Leibler divergence

β is 0.1. We use the Adam optimizer with a learning rate of $1e-3$ and train CNet for 150 epochs. The full training of a subject-specific model takes approximately 10 minutes on an NVIDIA V100 GPU.

References

- [1] Thiemo Alldieck, Marcus Magnor, Bharat Lal Bhatnagar, Christian Theobalt, and Gerard Pons-Moll. Learning to reconstruct people in clothing from a single RGB camera. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2
- [2] Bharat Lal Bhatnagar, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. Combining implicit function learning and parametric models for 3d human reconstruction. In *European Conference on Computer Vision (ECCV)*. Springer, 2020. 2
- [3] David Eigen, Marc’Aurelio Ranzato, and Ilya Sutskever. Learning factored representations in a deep mixture of experts. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Workshop Track Proceedings*, 2014. 2
- [4] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34, 2015. 2
- [5] Yu Rong, Takaaki Shiratori, and Hanbyul Joo. Frankmocap: A monocular 3d whole-body pose estimation system via regression and integration. In *IEEE International Conference on Computer Vision Workshops*, 2021. 2
- [6] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. <https://github.com/facebookresearch/detectron2>, 2019. 2
- [7] Sirui Xu, Zhengyuan Li, Yu-Xiong Wang, and Liang-Yan Gui. Interdiff: Generating 3d human-object interactions with physics-informed diffusion. In *ICCV*, 2023. 1