Hand-Object Contact Consistency Reasoning for Human Grasps Generation Supplementary Material

Hanwen Jiang[*] Shaowei Liu[*] Jiashun Wang Xiaolong Wang
UC San Diego
Project Page: https://hwjiang1510.github.io/GraspTTA/

Appendix

We provide more detailed information in the Appendix, including:

- Network architectures;
- Experiment and evaluation details;
- More results with visualization.

Appendix A: Network Architectures

We show structures of GraspCVAE and ContactNet in the following sections.

A.1. GraspCVAE

Stage	Configuration	Output
0	Input Hand point cloud $\mathcal{P}^h$ Input Object point cloud $\mathcal{P}^o$	778×3 3000×3
	3D Feature extraction	
1	Extract feature with two PointNet encoders	1024 ($\mathcal{F}^h$) 1024 ($\mathcal{F}^o$)
	Calculating posterior distribution (Input *concat*($\mathcal{F}^h$, $\mathcal{F}^o$))	
2	CVAE encoder (fc-layers, 2048, 1024, 512, 256, 64)	64 (μ) 64 (σ^2)
	Latent code sampling	
3	Sampling from calculated Gaussian	64 (z)
	Hand mesh reconstruction (Input *concat*($\mathcal{F}^o$, z))	
4	CVAE decoder (fc-layers, 1088, 1024, 256, 61)	61 (param)
4	MANO Layer	Hand mesh $\hat{\mathcal{M}}$

Table 1: Training time GrasCVAE architecture.

Table 1 and Table 2 show the architecture of GraspCVAE during training and testing respectively. The input of the two phase are different. During training, the input is both of hand and object point cloud and we train the network in a hand reconstruction manner. During testing, the only input is the object point cloud, and the network generates human hand mesh for grasping the object.

For training, we use two PointNet encoders to get features of hand and object point cloud as $\mathcal{F}^h$ and $\mathcal{F}^o$. Then, they are concatenated and sent to the CVAE encoder for predicting the posterior distribution $Q(z|\mu, \sigma^2)$. Then, a latent code z is sampled from this distribution, and concatenated with the object feature $\mathcal{F}^o$ as the input of CVAE decoder for regressing the MANO parameters. In the end, the parameters pass the MANO layer, where the output is the generated hand mesh $\hat{\mathcal{M}}$.

Stage	Configuration	Output
0	Input Object point cloud $\mathcal{P}^o$	3000×3
	3D Feature extraction	
1	Extract feature with a PointNet encoder	1024 ($\mathcal{F}^o$)
	Latent code sampling	
2	Random sampling in standard Gaussian	64 (z)
	Grasp Prediction (Input *concat*($\mathcal{F}^o$, z))	
3	CVAE decoder (fc-layers, 1088, 1024, 256, 61)	61 (param)
3	MANO Layer	Hand mesh $\hat{\mathcal{M}}$

Table 2: Test-time GrasCVAE architecture.

For testing, the latent code z is randomly sampled from the standard Gaussian distribution. Thus, we do not need the CVAE encoder and the hand point cloud.

A.2. ContactNet

Stage	Configuration	Output
0	Input Hand point cloud $\mathcal{P}^h$ Input Object point cloud $\mathcal{P}^o$	778×3 3000×3
	3D Feature extraction	
1	Extract feature with two PointNet encoders	1024 ($\mathcal{F}^h$) 1024 ($\mathcal{F}^o_g$) 3000×64 ($\mathcal{F}^o_l$)
	Feature fusion	
2	*concat*(*add*($\mathcal{F}^h$, $\mathcal{F}^o_g$).*repeat*(3000), $\mathcal{F}^o_l$)	3000×1088
	Contact map regression	
3	1-D convolutions (1088, 512, 256, 128, 1, sigmoid)	3000×1

Table 3: ContactNet architecture.

Table 3 shows the architecture of ContactNet, which takes in both hand-object point cloud to regress the object contact map. In the network, we use both of the object global and local features, $\mathcal{F}^o_g$ and $\mathcal{F}^o_l$, where the local features are used to maintain the point correspondence.

Appendix B: Details of Experiments and Evaluation

B.1. Datasets

We follow [2] to use HO-3D and FPHA datasets for evaluating the generalization ability of the proposed method. For FPHA dataset, the ground-truth hand mesh are fitted on the provided hand joints. We follow [2] to exclude the huge objects (especially milk bottle) in the FPHA dataset.

B.2. Evaluation Metrics

Perceptual score. The perceptual score is evaluated with Amazon Mechanical Turk following [2], the layout is shown in Fig. 1. We show 3 views of each sample. The rating score ranges from 1 to 5. Every sample is rated by 3 workers.

Penetration. The penetration is to measure the collision between the hand and the object. We report the maximum penetration depth and penetration volume following [1]. The former is calculated as the largest distance from the penetrating vertices of hand mesh to the closed object surface. And the latter is the volume of the intersecting voxels between the hand and object meshes. To compute this metric, we first voxelize both hand and object mesh using the voxel size of 0.5 cm, and then compute the number of intersecting voxels. The result is computed by the voxel volume times the number of intersecting voxels.

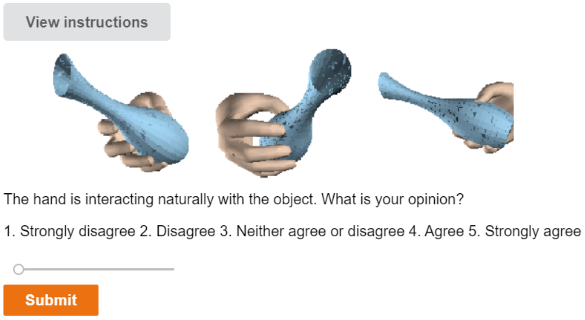
Reconstruction Error. We **do not** use hand reconstruction error (mesh reconstruction error on hand mesh, or kinematics error on hand joints) as a metric for evaluating the quality of generated grasps. Because grasp generation has multiple solutions, a good and reasonable grasp can be far away from the GT (Note that only one GT grasp is provided for each sample in datasets we used). Thus it does not make sense in our case to measure the reconstruction error with only one GT.

B.3. Experiments

We introduce more experiments details and results in this section, including details of GraspCVAE training targets and Test-time Adaptation.

B.3.1. GraspCVAE Training Targets

Table 4 shows the performance of the GraspCVAE trained by losses we proposed and losses from [3], which are tested on the Obman test set. Our training targets performs significantly better.



The hand is interacting naturally with the object. What is your opinion?

1. Strongly disagree 2. Disagree 3. Neither agree or disagree 4. Agree 5. Strongly agree

Figure 1: AMT online evaluation layout.

	Penetr Vol. ↓	Stability ↓	Percep Score ↑	Contact (%) ↑
[3]	8.41	1.66	2.97	98.25
Ours	**5.12**	**1.52**	**3.54**	**99.97**

Table 4: Performance of the GraspCVAE trained by losses we proposed and losses from [3] on Obman dataset.

B.3.2. Test-time Adaptation

Details of TTA During TTA, each test sample is adapted in a self-supervised manner for 10 iterations. For each iteration, the single test object is augmented into a batch which includes 32 samples, where the augmentation is random translation in $[-5, 5]$ cm. Due to the reason that the augmentation is supposed to maintain the geometry feature of the object, other augmentation methods, e.g. scaling and rotation, are harmful.

Details of Different TTA Paradigms In the Sec. 4.5.3 of the paper, we compare different TTA paradigms. And we give more details here.

- TTA-optm (offline): In this method, we only optimize the 45-D hand joints axis-angle rotation tensor, rather than the 61-D full hand pose parameters as in other learning-based TTA. We observe that optimizing the 61-D full hand pose is not stable, and the results can even become worse.

- TTA-noise (offline): In this method, when we train the ContactNet, we injecting random noise on the input 45-D hand joint rotation tensor. The model is denoted as ContactNet-noise. The reconstruction error of ContactNet-noise is **0.109**, higher than the original **0.090** without injecting noise (Table **??** in paper). The increased error demonstrate that injecting noise is harmful for learning contact maps, and implies that the network cannot learn to "corret" the noise. It is also the reason for the worse results of TTA-noise compared with original TTA.

- TTA-online: The HO-3D and FPHA datasets are video datasets, and the TTA-online is performed on the video clips. Because the object pose changes smoothly in the video frames, it provides the chance for the network

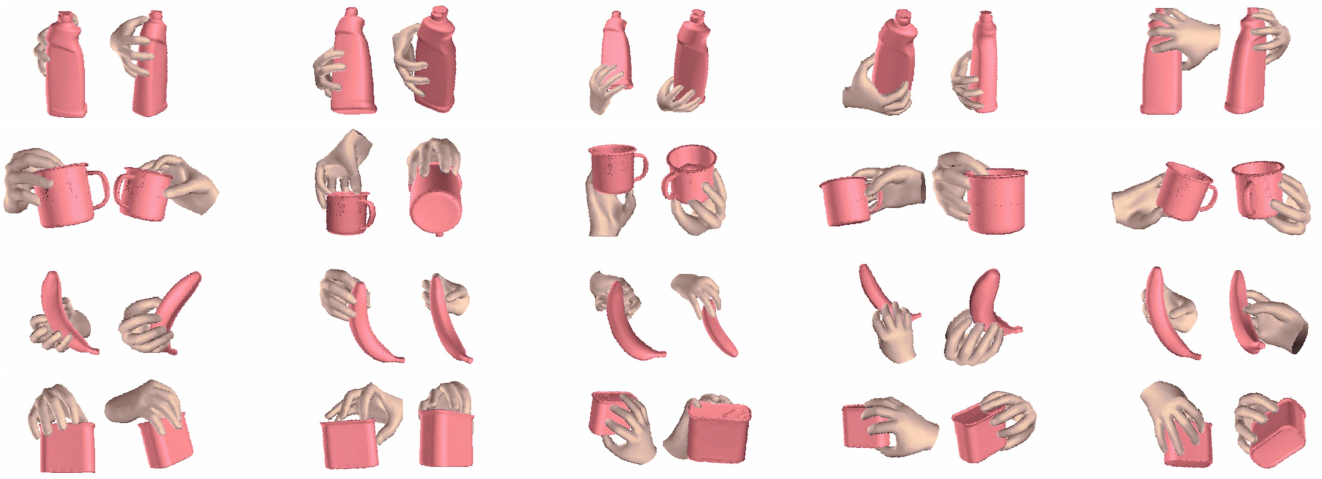


Figure 2: Diversity of generated grasps with examples on four out-of-domain objects. We show 5 results for each object, where each example is shown in 2 views.

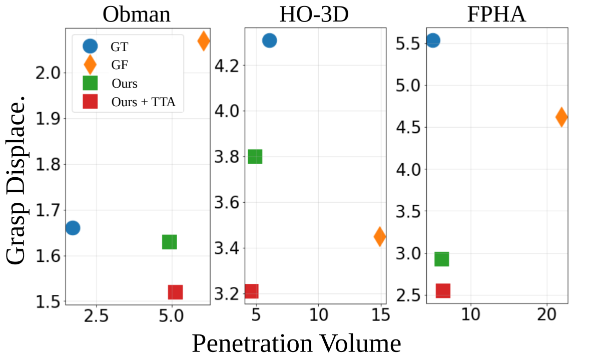


Figure 3: Balance between penetration volume (x-axis) and grasp stability (y-axis, measured by simulation displacement) on the three datasets, compared with ground truth (GT) and Grasping Field (GF) [2]. Results close to the origin point are better, indicating a grasp has better stability (smaller simulation displacement) with smaller penetration.

to fit the test distribution continuously better. Besides, the TTA-online also demonstrate that the model after TTA **does not overfit** to the single test sample, because it can continually generates grasps of the following incoming test samples without re-initializing the network parameters.

Appendix C: Additional Results

Diversity of generated grasps are shown in Fig. 2. By sampling different object poses for the same object as inputs, our model can generate diverse grasps. We show 5 different grasps generated by our model for each object in each row.

Penetration Volume vs. Grasp Displace. Larger penetration volume can cause better grasp stability (reflected by smaller simulation displacement) during the simulation. However, ideal grasps should be with small penetration and simulation displacement simultaneously, rather than achieving reasonable stability by suffering from huge penetration volume. Thus, we draw Fig. 3 for demonstrating the balance between them on the three datasets. Overall, our results are very close the origin point, which demonstrated our generated grasps has both small penetration and superior stability at the same time. With TTA, the results move vertically in the figure, indicating the TTA is able to increase grasp stability without magnifying the penetration at the same time. Besides, the results are comparable to or even outperform the ground truth.

More visualization are shown in Fig. 4 for in-domain Obman test set objects, and Fig. 5 for out-of-domain HO-3D objects. Each result is shown in a row. All results are chosen randomly.

More results are shown in Fig. 6 for in-domain Obman test set objects, and Fig. 7 for out-of-domain HO-3D and FPHA objects. We show 4 examples in each row, and each result is shown with 3 views. All results are chosen randomly.

References

[1] Yana Hasson, Gul Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *CVPR*, pages 11807–11816, 2019. 2

[2] Korrawe Karunratanakul, Jinlong Yang, Yan Zhang, Michael Black, Krikamol Muandet, and Siyu Tang. Grasping field: Learning implicit representations for human grasps. *arXiv preprint arXiv:2008.04451*, 2020. 2, 3

[3] Omid Taheri, Nima Ghorbani, Michael J Black, and Dimitrios Tzionas. Grab: A dataset of whole-body human grasping of objects. In *European Conference on Computer Vision*, pages 581–600. Springer, 2020. 2

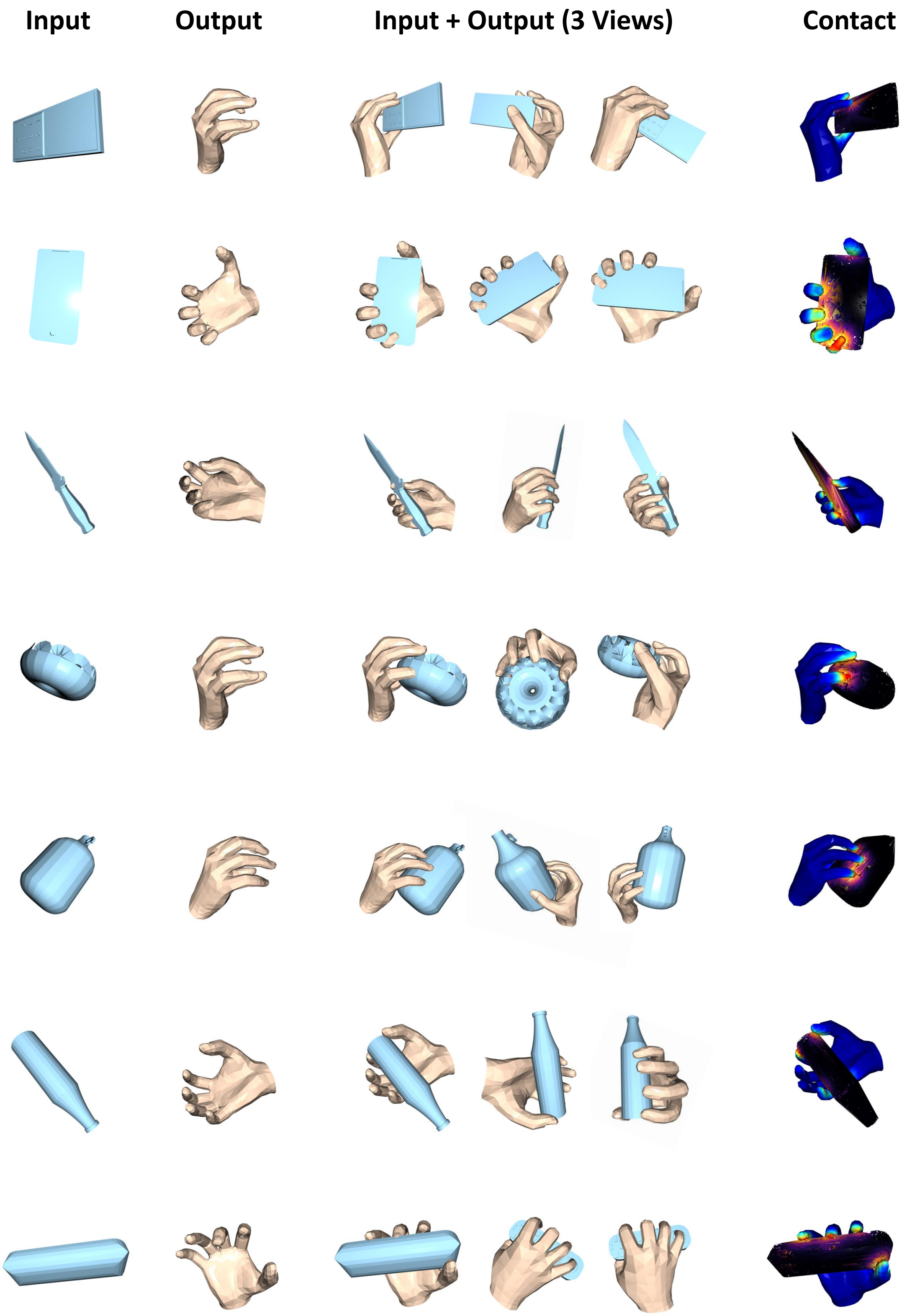


Figure 4: Results of generated grasps given in-domain objects. Every result is shown in a row with input object, output hand mesh, both input and output in 3 views and in contact.

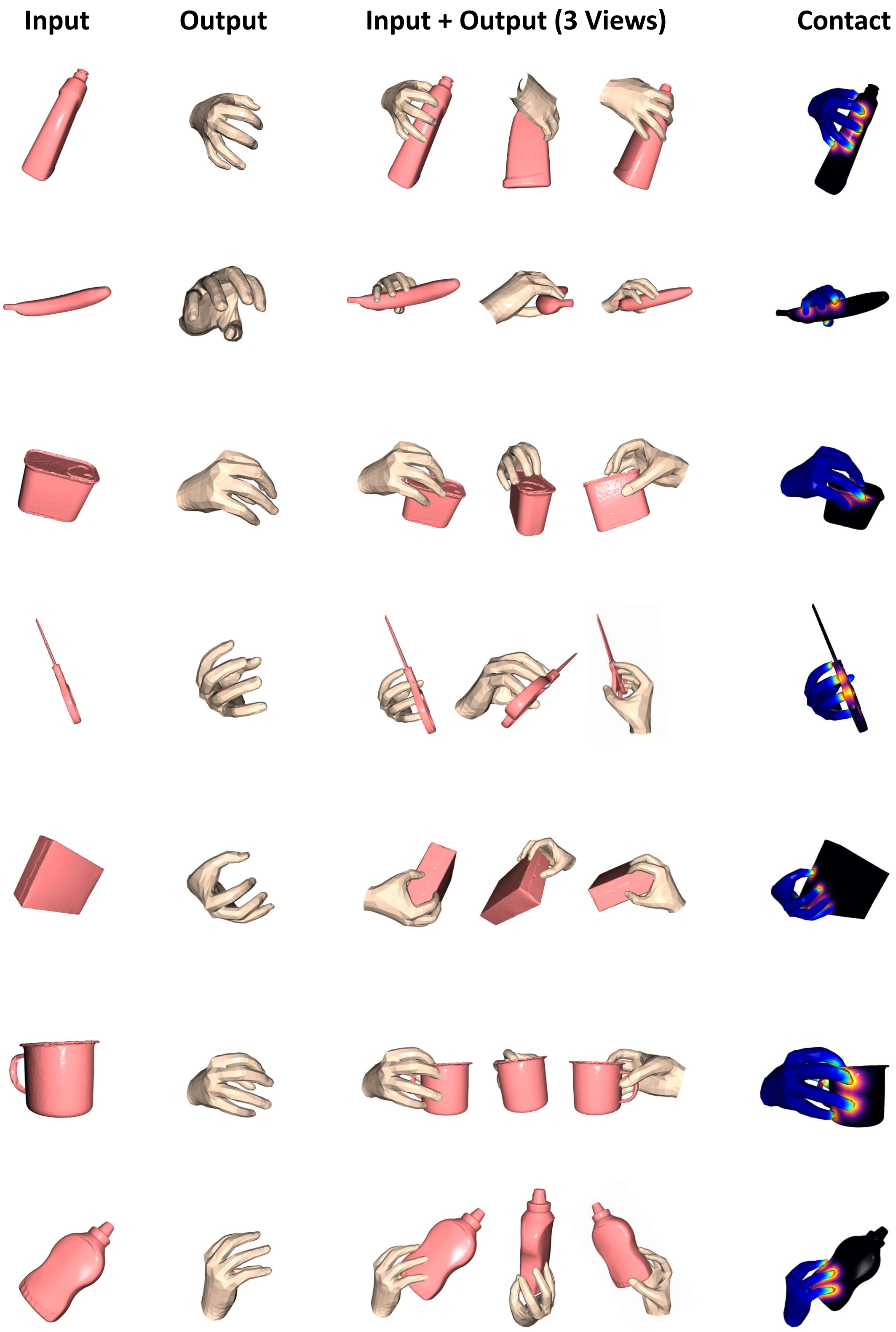


Figure 5: Results of generated grasps given out-of-domain. Every result is shown in a row with input object, output hand mesh, both input and output in 3 views and in contact.

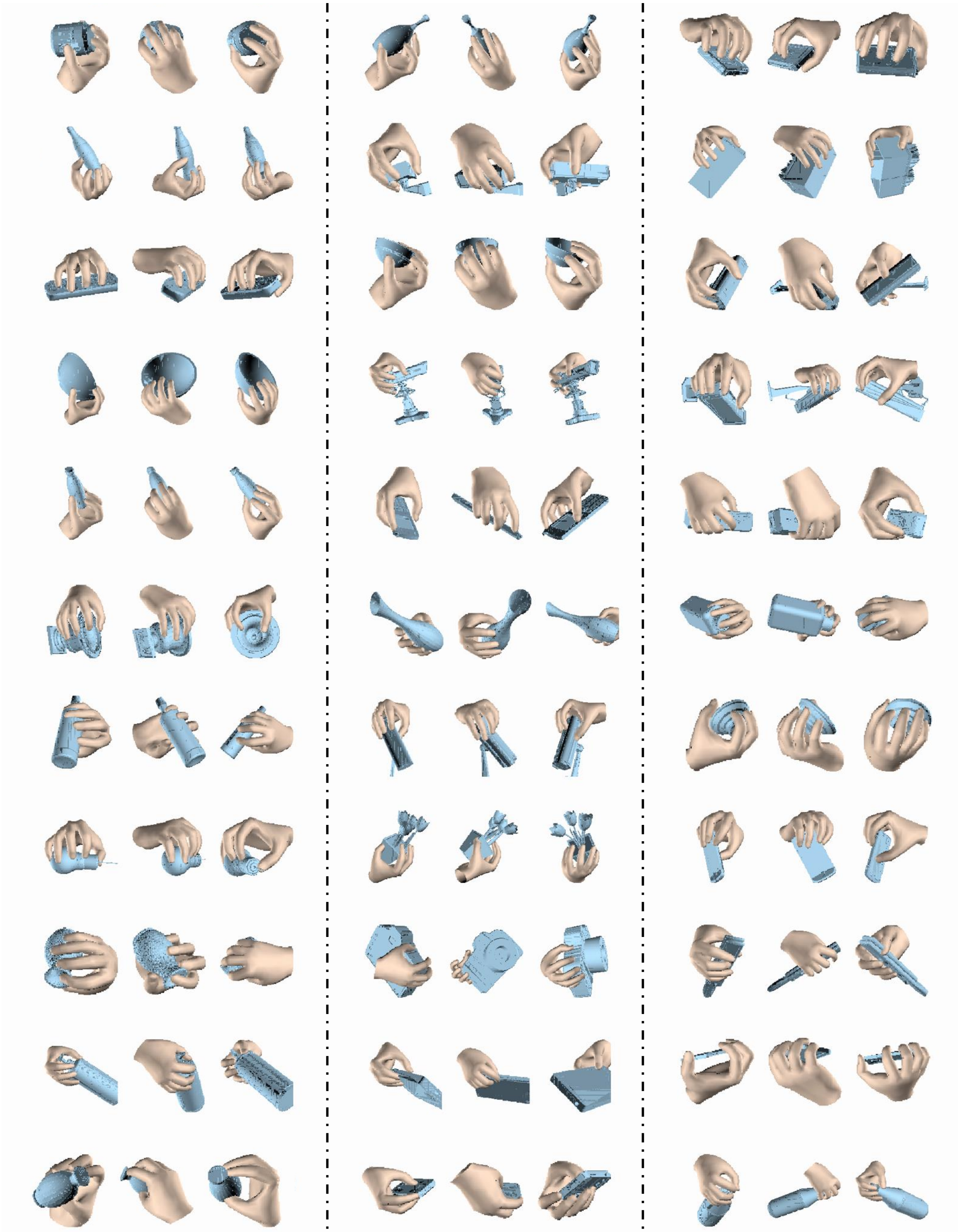


Figure 6: Generated grasps given in-domain Obman test objects. Each results is shown in 3 views. All results are chosen randomly.

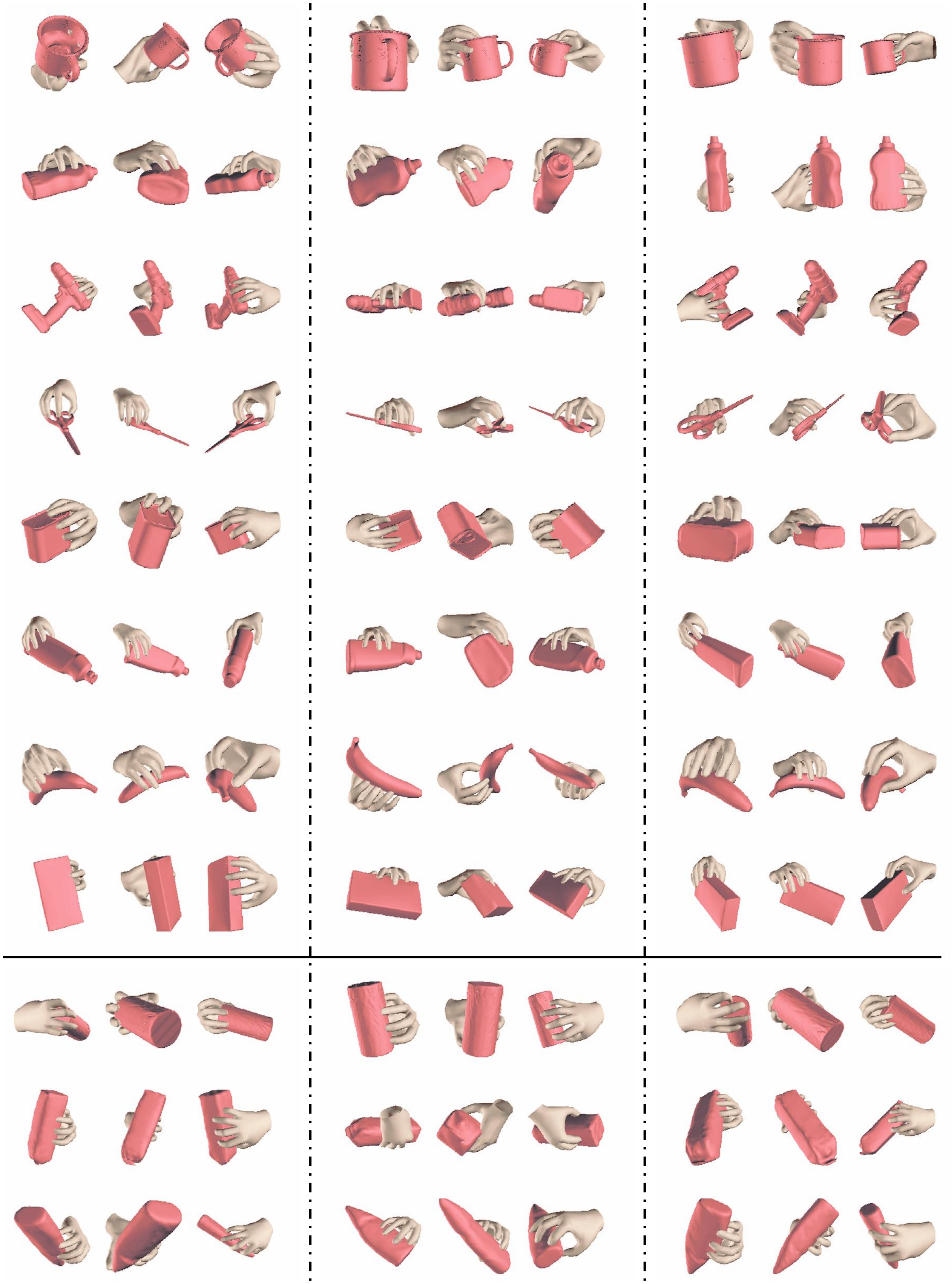


Figure 7: Generated grasps on out-of-domain HO-3D and FPHA objects. 8 out of 10 objects of HO-3D dataset and all 3 objects of FPHA dataset are visualized. We include 3 results of each object in each row. Each result is shown in 3 views. All results are chosen randomly.