Open Peer Review on Qeios

RESEARCH ARTICLE

FastGrasp: Efficient Grasp Synthesis with Diffusion

Xiaofei Wu[1], Tao Liu[1], Caoji Li[1], Yuexin Ma[1], Yujiao Shi[1], Xuming He[1]

1 ShanghaiTech University, Shanghai, China

Funding: This work was supported by NSFC 62350610269, Shanghai Frontiers Science Center of Human- centered Artificial Intelligence, and MoE Key Lab of Intelligent Perception and Human-Machine Collaboration (ShanghaiTech University).

Potential competing interests: No potential competing interests to declare.

Abstract

Effectively modeling the interaction between human hands and objects is challenging due to the complex physical constraints and the requirement for high generation efficiency in applications. Prior approaches often employ computationally intensive two-stage approaches, which first generate an intermediate representation, such as contact maps, followed by an iterative optimization procedure that updates hand meshes to capture the hand-object relation. However, due to the high computation complexity during the optimization stage, such strategies often suffer from low efficiency in inference. To address this limitation, this work introduces a novel diffusion-model-based approach that generates the grasping pose in a one-stage manner. This allows us to significantly improve generation speed and the diversity of generated hand poses. In particular, we develop a Latent Diffusion Model with an Adaptation Module for object-conditioned hand pose generation and a contact-aware loss to enforce the physical constraints between hands and objects. Extensive experiments demonstrate that our method achieves faster inference, higher diversity, and superior pose quality than state-of-the-art approaches. Code is available at https://github.com/wuxiaofei01/FastGrasp.

Corresponding authors: Yujiao Shi, shiyj2@shanghaitech.edu.cn; Xuming He, hexm@shanghaitech.edu.cn

Figure 1. FastGrasp provides extensive realistic grasping of dexterous hands synchronized with human poses.

1. Introduction

The problem of modeling hand-object interactions[1][2][3][4][5][6] has attracted increasing research interest recently, with important applications in virtual reality[7], human-computer interaction[8][9], and imitation learning in robotics. A key task in hand-object interaction modeling is to predict various ways a human hand can grasp a given object. Unlike robot grasping with parallel jaw grippers, the task of predicting human grasps is particularly challenging due to two reasons: First, human hands have more degrees of freedom, resulting in more intricate contact patterns; Moreover, the generated grasp must be not only physically plausible but also appear natural, reflecting the typical ways that humans handle objects.

Previous methods for synthesizing human grasping postures often rely on a two-stage process[10][11][12][13]. Such a process typically first uses a generative model, e.g., Conditional Variational AutoEncoder (CVAE)[14], to generate a series of intermediate representations, including contact maps[13] and/or parts maps[12], based on the point cloud representation of interacting objects. The second stage then uses those intermediate representations to estimate the hand parameters, aiming to produce a natural and physically plausible hand pose. To achieve this, most methods formulate the estimation as an optimization problem and adopt an iterative procedure to search the target hand pose. Despite their promising results, such two-stage methods often suffer from two drawbacks: First, the iterative optimization procedures are computationally intensive, leading to a low inference efficiency and time-consuming generation; Second, the quality of generated hand poses highly relies on the intermediate representations from the first stage and prone to accumulated errors.

To address those limitations, we propose an efficient one-stage generation method, named*FastGrasp*, to directly generate grasping poses without producing intermediate representations like contact maps, while maintaining the diversity of generated poses. To achieve this, we leverage the latent diffusion model framework[15] to learn a contact-aware representation for hand poses in a latent space and a diffusion-based generation process, capable of better encoding the physical constraints and capturing the object-conditioned hand-pose distribution.

Specifically, *FastGrasp* first learns a low-dimensional latent representation of hand pose parameters based on an AutoEncoder (AE) network. It then encodes the object with a Point-Net and builds a diffusion model in the latent space conditioned on the object representation. Subsequently, to incorporate the physical constraints on hand-pose interaction, *FastGrasp* introduces an adaptation module, which refines the diffusion-generated latent representation based on the object contact information. Finally, the contact-aware hand-pose presentation is decoded into the MANO[16] parameters of the grasping hand pose with the AE decoder.

We validate our approach through extensive experiments on three hand-object interaction benchmarks: HO-3D[17], OakInk[18], and Grab[19]. Experimental results demonstrate that our method achieves low latency in inference and generates higher-quality grasping poses with more plausible physical interactions and higher diversity than recent state-of-the-art approaches.

In summary, our contributions are as follows:

- We introduce FastGrasp, a diffusion-based one-stage model for generating grasping hand pose without requiring expensive iterative optimization.
- We propose an adaptation module to effectively incorporate physical constraints into a latent hand representation.
- Our approach achieves fast inference and outperforms previous state-of-the-art methods on a range of metrics.

2. Related Work

2.1. Hand-object Interaction

Generating whole-body interactions, such as approaching and manipulating static[20][21] and dynamic objects[22], is a growing topic. The task of synthesizing humans interacting with dynamic objects is explored using first-person vision[23] in skeleton-based datasets. However, numerous studies begin to explore hand-object interactions across diverse settings[24][25][26][4]. Most current efforts focus on synthesizing these interactions in the domains of computer graphics[27][28][29], computer vision[30][31][32][33][34][35][36], and robotics[37][38][39]. To perform hand-object pose estimation, Tekin *et al.*[40] proposes a 3D detection framework that predicts hand-object poses using two output grids without explicitly modeling their interaction. In contrast, Hasson *et al.*[41] utilize hand-centric physical constraints to model hand-object interactions and prevent penetration. Recently, research shifts towards generating plausible hand grasps for objects, with significant contributions including:[42][19]. GanHand[42] generates grasps suitable for each object in a given RGB image by predicting a grasp type from grasp taxonomy and its initial orientation, then optimizing for better contact with the object. GrabNet[19] represents 3D objects using Basis Point Set to generate MANO[16] parameters. The predicted hand is refined using an additional model to enhance contact accuracy. Our diffusion-model-based pipeline directly generates the grasping pose for a given object point cloud, eliminating the need for additional models.

2.2. Grasp Synthesis

Grasp synthesis receives extensive attention across robotic hand manipulation, animation, digital human synthesis, and physical motion control[21][43]. In this work, we focus on realistic human grasp synthesis[10][19][11][12][13], aiming to generate authentic human grasps for diverse objects. The key challenge is achieving physical plausibility and generation efficiency. Most existing approaches employ CVAE to generate hand MANO parameters[19][13][18] or hand joints[11]. Liu \etal.[12] propose learning intermediate representations followed by iterative optimization in two stages. This method weakens the spatial information of objects, causing intersection penetration and displacement, and requires significant time for optimization in the second stage. In contrast, we develop an one-stage generation model that supervises the spatial geometry of objects and adaptively learns the physical constraints of hand-object interaction. Such model architecture effectively accelerates generation speed and reduces hand-object penetration volume.

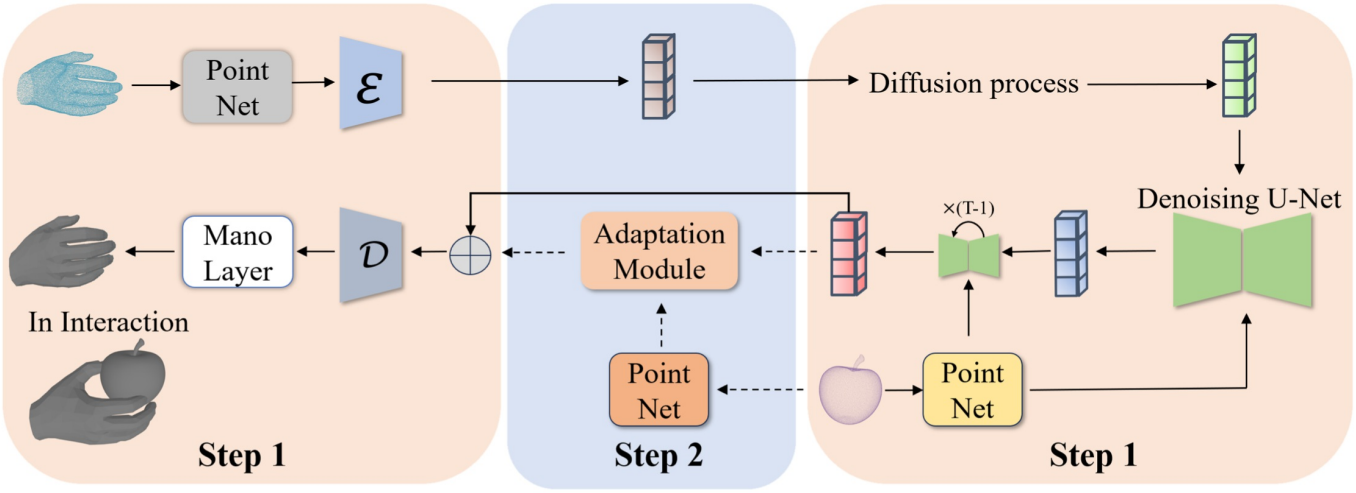

Figure 2. Model training architecture. We divide the training process into two parts. In the first part, we use a latent diffusion model to generate grasping poses from object point clouds. However, the diffusion model struggles to directly learn the physical constraints between the hand and object, leading to issues such as penetration and displacement. To address this, the second part involves training an Adaptation Module to refine the grasping gestures by aligning them with the physical constraints of hand-object interactions, resulting in more natural and feasible poses. In training stage one, only the solid arrow path is utilized. In stage two, both the solid and dotted arrow paths are used.

2.3. Denoising Diffusion Probabilistic Models

Denoising diffusion models[44][45][46][47][48] utilize a stochastic diffusion process that incrementally introduces noise into a sample from the data distribution, adhering to thermodynamic principles. They then generate denoised samples through a reverse iterative procedure. However, directly training DDPMs on high-resolution point clouds and sampling from them is computationally intensive. Latent diffusion models address this issue by encoding high-resolution images into a low-dimensional latent space[49][50][15] before training DDPMs. Our approach follows this paradigm: we first train an autoencoder in the data space, and then train a DDPM using the encoded samples. Additionally, we designe an Adaptation Module(AM) to adjust the input to the decoder, incorporating hand-object physical constraints into the diffusion model.

3. Fast Grasping Hand Pose Generation

3.1. Method Overview

Given an object, usually represented by a point cloud, our purpose is to generate a human hand pose for grasping this object. The generated grasping hand pose should be natural and physically correct, securely holding the object in a physically plausible manner. Unlike the existing methods that usually adopt a two-stage design with high computation cost, we propose *FastGrasp*, a fast grasping hand pose generation pipeline without estimating intermediate representations and iterative optimizations.

FastGrasp is a one-stage generation framework consisting of two main modules for generating the grasping hand pose.

The first module is based on a latent diffusion model to preserve the diversity of hand poses when intermediate representations like contact maps are absent. Given the latent hand representation generated from the diffusion model, we introduce an adaptation module to enforce the physical constraints of hand-object interaction. This design allows the model to directly learn the spatial relationship between the hand and object point clouds without iterative optimization, resulting in a fast generation of high-quality hand poses.

To learn the entire model, we adopt a simple yet effective two-step training strategy. The first step trains the latent diffusion model, which generates an initial representation of the hand poses. Next, we train the adaptation module to refine the hand representation to strengthen the physical constraints of the hand-object interaction. After training, our generation requires only one pass of network inference, thus significantly accelerating grasping hand generation.

Below we will first introduce the latent diffusion model module in Sec. 3.2, followed by the adaptation module in Sec. 3.3. Finally, the model inference pipeline will be detailed in Sec. 3.4.

3.2. Latent Diffusion Model for Hand Pose

Latent Hand Representation

To build our Latent Diffusion model[50] for hand pose, we first train an auto-encoder that maps the input hand representation to a latent space. This allows us to reduce the data dimensionality for the diffusion process and improves the modeling efficiency. In contrast to the original latent diffusion model, where the input and output are exactly the same, we employ an asymmetric design in the auto-encoder for the subsequent conditional generation process.

Specifically, the input to our auto-encoder is the vertices of the hand mesh,$h_v \in \mathrm{R}^{778\times 3}$, which is first processed by a PointNet[51] and then fed into the encoder block. This design maintains the spatial shape information of the input hand in feature extraction, which can be more easily integrated with the object representation in the later stage. The obtained latent vector is converted to MANO[16] parameters representation $h_p \in \mathrm{R}^{61}$ instead of the vertices by the decoder block. The MANO parameters have far less freedom than those of vertices, thus improving the regularization in learning the decoder. The hand mesh vertices $h_m \in \mathrm{R}^{778\times 3}$ is finally reconstructed from h_p by a differentiable MANO layer[16].

The training objective of the AutoEncoder combines a hybrid reconstruction loss and a set of physical constraints. The reconstruction loss measures the difference between the reconstructed hand mesh and the ground truth, which includes two terms:

$$\mathrm{L}_{recon} = \lambda_1 \mathrm{L}_{param} + \lambda_2 \mathrm{L}_{mesh}$$

$$\mathrm{L}_{param} = \mathrm{MSE}(h_p, h_p^{gt})$$

$$\mathrm{L}_{mesh} = \text{Chamfer-Dis}(h_m, h_m^{gt})$$

where L_{param} indicates mean squared error loss between predicted h_p and GT hand MANO parameters h_p^{gt}, L_{mesh} measures chamfer distance between the predicted hand vertices h_m and the GT hand vertices h_m^{gt}. λ_1 and λ_2 are the

weight balancing coefficients.

To learn a hand representation that adheres to physical constraints, we also employ the following three loss functions from[13]:

$$L_{consist} = \text{Consist}(h_m, h_m^{gt}, o_m)$$

$$L_{cmap} = \text{Contact}(h_m, o_m)$$

$$L_{penetr} = \text{Penetra}(h_m, o_m)$$

where o_m denotes the object mesh that we aim to grasp,$L_{consist}$ aims to make the contact region of the predicted hand mesh on the object consistent with that of the GT hand mesh on the object. L_{cmap} ensures that the hand mesh generated by the model maintains contact with the object. L_{penetr} prevents the hand mesh and objects from penetrating the physical volume. We refer the reader to the Supplementary for details of those loss functions.

Our total loss function for training the auto-encoder (the left part in Fig. 2) can be written as:

$$L = L_{recon} + \lambda_3 L_{cmap} + \lambda_4 L_{penetr} + \lambda_5 L_{consist}$$

where $\lambda_3, \lambda_4, \lambda_5$ are weight parameters for balancing the physical constraint loss terms. By integrating physical and reconstruction losses, our model is able to learn the hand mesh and the physical constraints involved in the interaction between the hand and the object. This approach ensures that our auto-encoder effectively encodes the hand vertices and maintains the physical plausibility of the generated mesh.

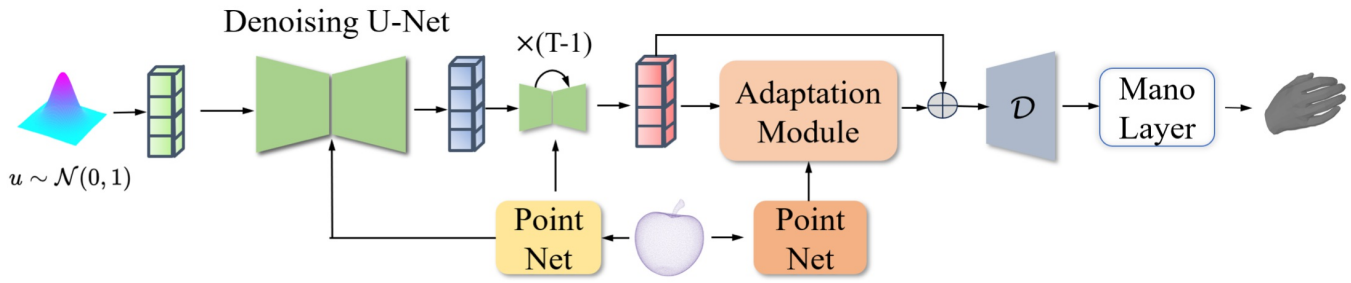

Figure 3. Model inference architecture. We start by inputting Gaussian noise and the object's point cloud into the model. The diffusion model then generates hand representations in latent space. The Adaptation Module refines these representations, which are then decoded into MANO parameters. Finally, we construct the hand mesh using the MANO layer.

Diffusion Model for Hand Representations

We adopt a diffusion model to learn the distribution of the latent hand representation produced by the auto-encoder. The model gradually denoises a normally distributed random variable, which corresponds to learning the reverse process of a fixed Markov Chain[44][52]. Here we train a denoising U-Net to predict the added noises in the diffusion process, as shown in the right part of Fig. 2.

Specifically, the input of the diffusion model consists of three parts:z_0, $o_p \in R^{N_o \times 3}$ and t. z_0 be the feature output of the encoder E when the input is h_v. The input object point cloud o_p, is used as the conditioning information for our diffusion model. It is transformed into an embedding using PointNet[51], facilitating controllable generation. t denotes the time step in the diffusion model training process. The loss function for training the diffusion network can be written as:

$$L_{LDM} := \mathrm{E}_{\mathrm{E}(h_v), \epsilon \sim \mathrm{N}(0,1), t}\left[\left\| \epsilon - \epsilon_\theta(z_0^t, P(o_p), t) \right\|_2^2\right]$$

where $\epsilon_\theta(z_0^t, P(o_p), t)$ denotes the conditional denoising U-Net used for training, where t ranges from 1 to T, the input z_0^t is the z_0 mixed with ϵ, the P denotes the PointNet[51]. Through training, the diffusion model learns to reconstruct the hand mesh from Gaussian noise by denoising and decoding.

3.3. Physical Constraints Alignment

During the training of the diffusion model, directly incorporating physical loss and reconstruction loss lead to oscillations and hampers convergence. We attribute this issue to the diffusion model's difficulty in simultaneously learning the distribution of the E output and capturing the physical constraints between the hand and the object. Therefore, the generated hand mesh and object may exhibit significant physical penetration and displacement. To address this problem, we decompose the entire training process into a two-step optimization approach. This method not only simplifies the model's training complexity but also helps better capture the physical constraint relationship between the hand and the object.

Specifically, after training the diffusion model, we aim to adjust the physical constraints of hand-object interactions. To retain the knowledge from the previous diffusion model, we introduce an adaptation module f_{adapt} based on a MLP. The diffusion model's output z_1, serves as the input to the adaptation module. This module aligns the distribution learned by the diffusion model with the physical constraints of hand-object interactions. The specific formula is as follows:

$$z_2 = f_{adapt}(z_1)$$

where $z_2 \in R^{N_z}$, is the output of the adaptation module when given z_1 as input.

The goal of incorporating hand-object physical constraints is to ensure that the resulting hand mesh achieves natural and realistic grasping postures. However, z_1 and z_2 do not accurately represent the quality of hand-object interactions in real physical space. Therefore, we first reconstruct z_1 and z_2 back to the MANO parameters h_p, and then use the MANO Layer[16] f_{mano} to reconstruct the hand mesh h_m:

$$h_p = \mathrm{D}(z_1 + z_2)$$
$$h_m = f_{mano}(h_p)$$

Next, we update the adaptation module using the loss function 7 to ensure that the physical constraints of hand-object interactions are accurately aligned. This training method addresses the challenge of directly learning physical constraints in diffusion models, resulting in more natural grasping poses and minimizing unnecessary physical penetration.

3.4. Inference

Fig. 3 illustrates the inference process of our method. During inference, the initial input consists of noise u sampled from a Gaussian distribution and an object point cloud o_p.

First, we generate the prior z_1 for the hand mesh in the latent space through an N-step denoising process[53]. Next, the adaptation module integrates z_1 with the object point cloud information to generate z_2, as shown in Eq. 9. Finally, z_1 and z_2 are combined, and the decoder converts them into MANO parameters h_p, which are then used by the MANO layer[16] to produce the hand mesh h_m. This process can be described by the equations 10 and 11.

While using Diffusion Models (DDPM) for generating grasp postures marks a significant advancement over the previous two-stage model, there is still a need to enhance generation speed to meet practical requirements. To address this, we employ DDIM[53], which optimizes both speed and quality by adjusting the step size during the denoising process. This approach enables the rapid generation of grasping poses.

4. Experiment

Table 1. Ablation study results on the **GRAB, OakInk, HO-3D** datasets[19][18][17]. The evaluation of the HO-3D is an out-of-domain generalization test, where the model is trained using the GRAB dataset.

Dataset	Details	Penetration Volume ↓	Simulation Displacement ↓	Contact Ratio ↑	Entropy ↑	Cluster Size ↑
OakInk[18]	Baseline CVAE model	13.08	1.78	**98**	2.81	1.12
	Original diffusion model	18.34	**1.45**	**98**	2.91	**5.24**
	Original diffusion model with physical loss	6.31	3.77	71	2.85	1.58
	Our whole pipeline	**4.37**	**1.45**	94	**2.92**	4.96
GRAB[19]	Baseline CVAE model	12.33	1.94	98	2.62	0.87
	Original diffusion model	15.46	1.80	96	2.87	**3.06**
	Original diffusion model with physical loss	8.43	5.24	50	2.84	1.26
	Our whole pipeline	**1.25**	**1.67**	**100**	**2.93**	1.87
HO-3D[17]	Baseline CVAE model	23.17	3.12	**100**	2.64	0.93
	Original diffusion model	16.64	2.18	90	2.87	4.04
	Original diffusion model with physical loss	12.73	3.87	62	2.87	1.37
	Our whole pipeline	**5.23**	**2.14**	98	**2.88**	3.97

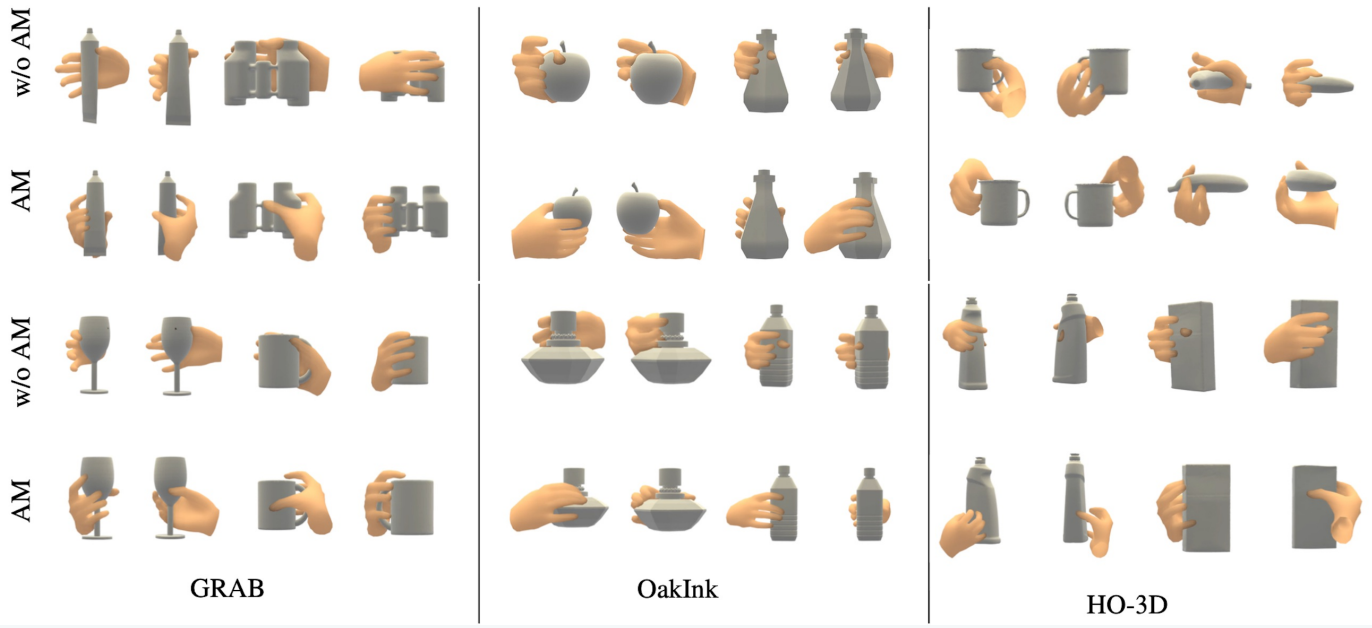

Figure 4. Qualitative comparison between our method and Ours w/o Adaptation Module (AM). Starting from the same random Gaussian noise, we visualize the generated grasps by our whole pipeline (first row) and ours w/o Adaptation Module. For each object, we show two different views for visualization (two columns). This comparison demonstrates that our whole pipeline with AM notably reduces object penetration and produces more realistic grasp poses.

GRAB[19]
OakInk[18]
HO-3D[17]

In this section, we evaluate the effectiveness and efficiency of the proposed framework for object-conditioned hand pose generation. The structure is organized as follows.

We first introduce our benchmarking datasets (Sec. 4.1), evaluation metrics (Sec. 4.2), and implementation details (Sec. 4.3). Then, we conduct a model analysis to demonstrate the efficacy of each component in the proposed framework (Sec. 4.4). In what follows, we compare our method with the recent state-of-the-art approaches (Sec. A2). Finally, we assess the perceived quality and stability of the generated grasping poses through user studies (Sec. 4.6).

For experimental settings, we assess the model's generalization to new objects using the out-of-domain dataset[17]. We also evaluate the physical penetration and grasp firmness of the generated poses with an in-domain setting on the OakInk and GRAB datasets[18][19].

4.1. Datasets

Table 2. Comparison with previous methods on the HO-3D dataset [17], where Ours[1] indicates our model is trained on the GRAB[19] dataset following[10][12][11], and Ours[2] and ContactGen[2] suggests the corresponding models are trained on the OakInk[18] dataset. Our model achieves state-of-the-art performance on this out-of-domain dataset, setting new benchmarks with faster inference speeds and the best physical metrics for generated grasps.

Method	Penetration Volume ↓	Simulation Displacement ↓	Contact Ratio ↑	Entropy ↑	Cluster Size ↑	Inference Time ↓
GrabNet[19]	15.50	2.34	99	2.80	2.06	0.23s
GraspTTA[13]	7.37	5.34	76	2.70	1.43	6.90s
HALO[11]	25.84	3.02	97	2.81	4.87	10.42s
GF[10]	93.01	-	**100**	2.75	3.44	32.75s
ContactGen[12]	9.96	2.70	97	2.81	**5.04**	110.60s
Ours[1]	**5.23**	**2.14**	98	**2.88**	3.97	0.14s
ContactGen[2]	14.32	2.41	**100**	2.84	**5.23**	110.60s
Ours[2]	**12.30**	**1.44**	**100**	**2.88**	4.41	**0.14s**

We conduct experiments using the OakInk[18], GRAB[19], and HO-3D[17] datasets, adhering to the experimental protocols outlined in[11][12][18]. Specifically, in Sec. 4.5, We train the model separately on the OakInk and GRAB datasets, and then evaluate its generalization ability on the HO-3D dataset. In Sec. 4.5, we perform both training and evaluation on the OakInk and GRAB datasets.

The OakInk and GRAB datasets[18][19] consist of hand-object mesh pairs with hand models parameterized by the MANO[16] model. The GRAB dataset includes real human grasps for 51 objects across 10 subjects, whereas the OakInk dataset features real human grasps for 1,700 objects from 12 subjects. Following[12][19][13], we also evaluate the model's generalization ability by testing on out-of-domain objects from the HO3D dataset.

4.2. Evaluation Metrics

Following the prior evaluation protocals[19][10][11][12][13][21], we evaluate the generated grasping poses using the following criteria: (1) physical plausibility, (2) stability, (3) diversity, (4) generation speed, and (5) perception score.

Physical Plausibility Assessment. We evaluate physical plausibility by measuring hand-object mutual penetration volume and contact ratio[10][11][12][13]. The penetration volume is calculated by voxelizing the mesh into $1 mm^3$ cubes and computing the overlapping voxels. The contact ratio indicates the proportion of the grasps in contact with the object.

Grasp Stability Assessment. Following[10][43][21][19][12][13][18], we use a simulator to position the object and the generated grasps. We then measure the average displacement of the object's center of gravity due to gravity.

Diversity Assessment. We assess the diversity of generated grasps following[11][12]. First, we cluster the grasps into 20

clusters using K-means. Diversity is measured by computing the entropy of cluster assignments and the average cluster size, with higher entropy values and larger cluster sizes indicating greater diversity. Consistent with previous work, K-means clustering[11][12] is applied to 3D hand keypoints across all methods.

Generation Speed Assessment. We randomly select 128 objects from the dataset, generate grasping poses for each object, and calculate the average time required to generate a single pose on an NVIDIA A40 GPU.

Perceptual Score Assessment. We conduct a perceptual study, as described in[11][13], with human participants to evaluate the naturalness of the generated grasps.

4.3. Implementation Details

During training, we use the Adam optimizer, LR $=1e^{-4}$, $N_z = 768$, $N_o = 3000$ and bath size = 256. During training the autoencoder, the loss weights are $\lambda_1 = 0.1, \lambda_2 = 1, \lambda_3 = 1000, \lambda_4 = 10, \lambda_5 = 10$. When training the diffusion model, we freeze the auto-encoder and sample 3000 points from the object mesh o_m as the input point cloud o_p. When training the adaptation module, we use the same input point cloud and the loss weights are
$\lambda_1^d = 100, \lambda_2^d = 0.1, \lambda_3^d = 1000, \lambda_4^d = 20, \lambda_5^d = 0.1$.

4.4. Ablation Study

In this section, we conduct an ablation study to systematically evaluate the contribution of each module to the overall framework performance. This approach clarifies the role and impact of each component before delving into a detailed analysis of the experimental results.

Tab. 1 summarizes the results, showing that while the CVAE model slightly outperforms the diffusion model in penetration rate, it exhibits weaker generative performance, as indicated by lower entropy and smaller cluster sizes. Conversely, the diffusion model excels in entropy and cluster size but struggles with higher penetration, suggesting difficulties capturing the physical constraints of hand-object interactions. Integrating a physical loss function directly into the diffusion model decreases performance by increasing displacement and reducing grasp robustness, underscoring the challenge of aligning hand representations with physical constraints in latent space. Our Adaptation Module approach effectively combines the diffusion model with physical constraints, achieving reduced penetration and displacement, and significantly improving the accuracy of hand-object interactions.

Fig. 4 shows that our Adaptation Module method significantly enhances performance across all three datasets, reducing penetration volume and improving generalization on the out-of-domain HO-3D dataset. This improvement further demonstrates the Adaptation Module's ability to transform distributions, aligning the generated hand latent vector with natural human expectations.

Table 3. Quantitative comparison on the OakInk and GRAB dataset [19][18], where * indicates the model is trained on the OakInk dataset using the code released by the authors. Our method achieves the best performance on almost all evaluation metrics.

Dataset	Method	Penetration Volume ↓	Simulation Displacement ↓	Contact Ratio ↑	Entropy ↑	Cluster Size ↑
OakInk[18]	GrabNet[18]	6.60	**1.21**	94	1.68	1.22
	ContactGen*	4.85	2.01	94	2.88	4.07
	Ours	**4.37**	1.45	**94**	**2.92**	**4.96**
GRAB[19]	GrabNet[19]	1.72	3.65	96	2.72	1.93
	HALO[11]	2.09	3.61	94	2.88	2.15
	ContactGen[12]	2.16	2.72	96	2.88	**4.11**
	Ours	**1.25**	**1.67**	**100**	**2.93**	1.87

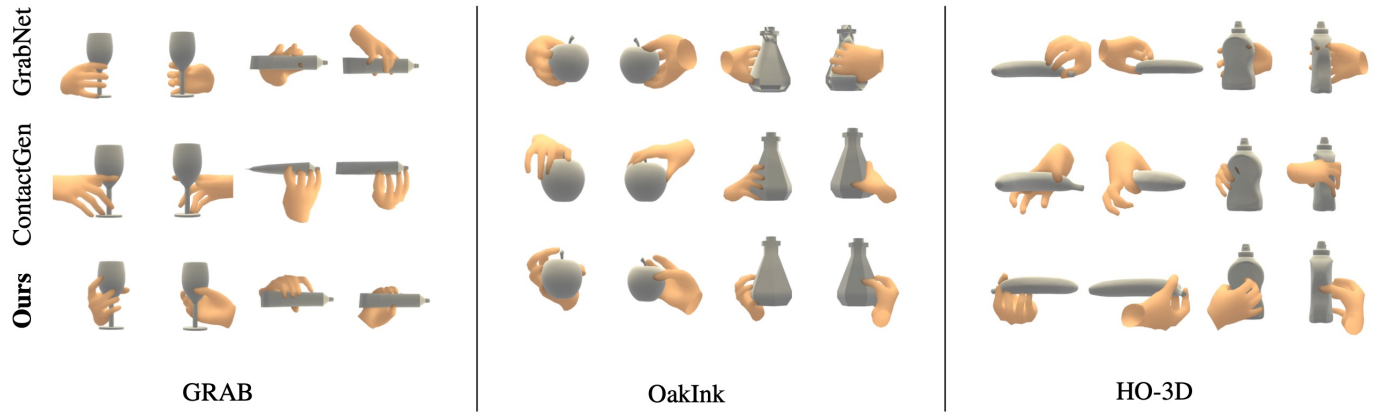

Figure 5. Qualitative comparisons with state-of-the-art methods on GRAB, OakInk, and HO-3D datasets. Each pair (two columns) visualizes the generated grasps from two different views. Our method demonstrates a significant reduction in object penetration compared to other methods.

GRAB[19]
OakInk[18]
HO-3D[17]

4.5. Grasp Generation Performance

Out-of-Domain. We assess the generalization ability of our model using the HO-3D dataset[17]. As demonstrated in Tab. 2 and Fig. 5, our method achieves the fastest generation speed, superior physical constraints, and entropy. In comparison, GrabNet[19] matches our method in generation speed but suffers from significant physical penetration. ContactGen excels in cluster size but has the longest generation time, making it impractical for real-world applications. Overall, our method outperforms previous approaches in both physical generalization and generation speed. **In-Domain.** Tab. 3 and Fig. 5 compare our method with ContactGen[12] and GrabNet[19] on the OakInk dataset. Our method excels in penetration, contact ratio, entropy, and cluster size. Although displacement is slightly higher than GrabNet, our method achieves significantly lower penetration volume, demonstrating a better balance between minimizing physical intrusion and

improving grasping effectiveness.

Tab. 3 compares our method with ContactGen[12], Halo[11], and GrabNet[19] on the GRAB dataset. Our approach outperforms the others by achieving the lowest penetration and displacement and the highest contact ratio. Fig. 5 demonstrates that our method produces highly plausible object grasping. Although ContactGen produces more diverse grasps than our method in terms of cluster size, our method archives better results with smaller penetration and greater stability. By focusing on detailed geometric spatial information, our model creates more precise grasping poses. This precision increases entropy for objects with varied geometries, leading to more diverse hand poses, while similar object geometries result in more uniform grips and lower cluster sizes.

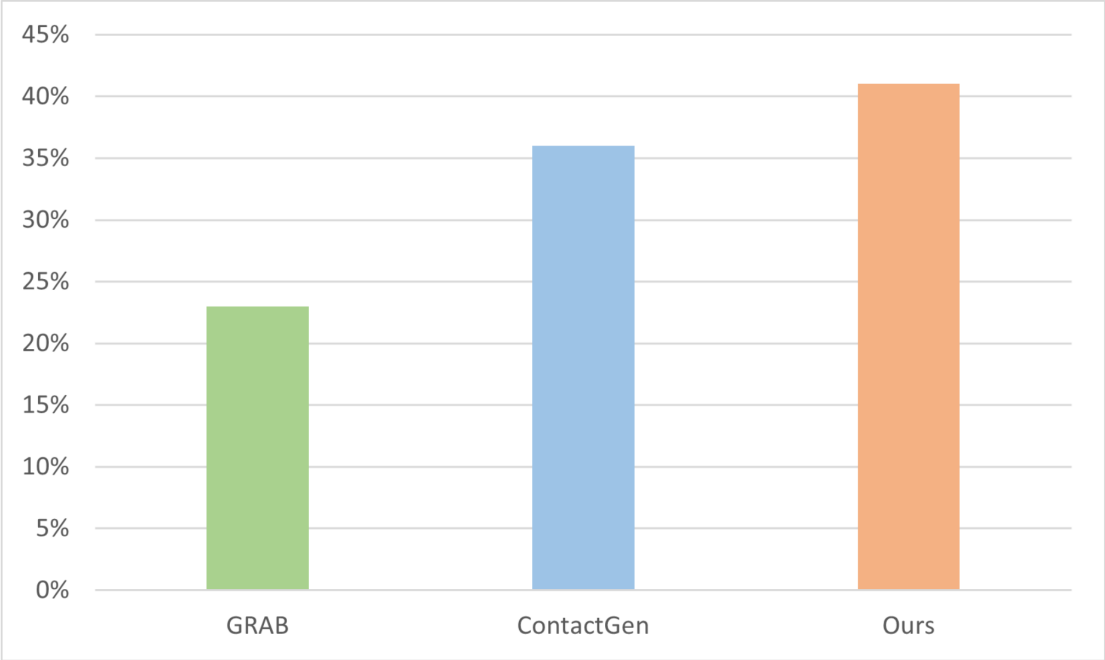

Figure 6. User study results. The numbers indicate the percentage of users who rate the corresponding method as more realistic.

4.6. User Study

We conduct a user study to evaluate the perceived quality and stability of grasps generated by different methods. We compare grasps generated by GrabNet[19], ContactGen[12], and our method by evaluating 10 objects from the GRAB[19], OakInk[18], and HO-3D[17] datasets. Each object is tested with 3 grasps from each method. Ten participants select the best grasp pose based on the naturalness and stability of the grasp. Fig. 6 shows that our method received the highest number of selections in the experiment, indicating it generates the most natural and stable grasps.

5. Conclusion

In this paper, we introduce a one-stage framework for rapid and realistic human grasp generation, eliminating the need for

iterative optimization processes common in previous methods. We introduce an adaptation module that aligns the generative model's output with physical constraints, refining hand representations in the latent space to enhance the accuracy and realism of generated grasps. Consequently, our method accelerates grasp generation, improves physical plausibility, and demonstrates robust generalization across diverse test inputs.

A. Supplementary Material

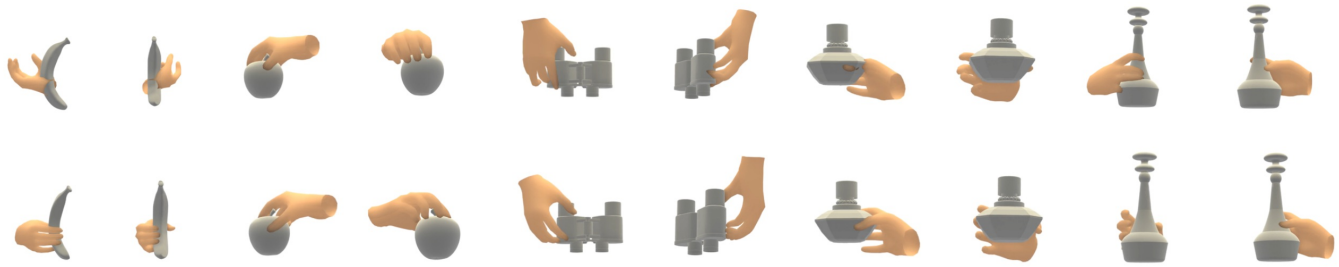

Figure 7. To assess the impact of a physically constrained loss function, we compare model performance with and without it. Each pair of columns shows generated grasps from two distinct views. The first row uses only the reconstruction loss, while the second row presents results from our proposed pipeline. Our method significantly reduces object penetration compared to using the reconstruction loss alone.

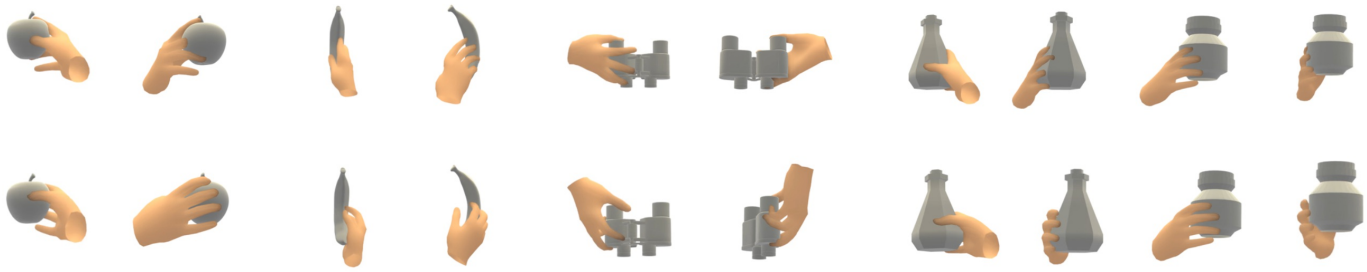

Figure 8. To evaluate the necessity of hand vertices as inputs, we visualize the model's output using both hand parameters and hand vertices. Each pair of columns shows generated grasps from two different views. The first row presents results with hand parameter input, while the second row displays results from our pipeline. Our method enhances performance by capturing hand joint details and improving rotational accuracy, which reduces object penetration.

Table 4. We conducted ablation experiments to evaluate the impact of the physical constraints loss function and hand vertices.

OakInk	Simulation Displacement ↓	Penetration Distance ↓	Penetration Volume ↓	Contact Ratio ↑
No-physical-loss	1.91	0.93	4.76	96
Hand param	**1.39**	**0.91**	5.91	**98**
Ours	1.83	**0.91**	**2.39**	**98**

A1. Overview of Material

The supplementary material comprehensively details our experiments, results, and visualizations. Tab. 4 examines the impact of physical constraints during autoencoder training and compares the effects of hand verts versus hand parameters as inputs. Sec. A2.3 offers additional visualizations to enhance understanding of our model.

A2. More Autoencoder Experimental Results

In training the autoencoder, we use hand vertices as input and apply both reconstruction and physical loss functions. Sec. A2.1 and Sec. A2.2 examine the effects of training the model with hand vertices and reconstruction loss alone versus using MANO parameters with both reconstruction and physical loss functions in Tab. 4.

A2.1. Training Using Reconstruction Loss

The model is trained using hand vertices h_v as input and relies solely on the reconstruction loss function, without incorporating any physical loss function. As shown in Fig. 7, experiments reveal that using only the reconstruction loss often results in significant penetration and displacement issues in hand-object interactions. However, as demonstrated in Tab. 4, incorporating a physical constraint loss function improves the model's ability to capture these details, reducing physical collisions and enhancing grasp stability.

A2.2. Training Using Mano Parameter

The model is trained using hand parameters h_p as input. Our experiments indicate that using hand vertices instead of MANO parameters results in less physical volume intrusion. As shown in Fig. 8 and Tab. 4, this is attributed to the Hand vertices providing a more robust data representation than MANO parameters, reducing the model's sensitivity to input variations and thus improving training effectiveness.

A2.3. Autoencoder Visulization Result

To validate the effectiveness of our autoencoder model, we provide extensive visualizations in Fig. 9 and 10.

Fig. 9 illustrates two grasping poses for randomly selected test objects. This demonstrates that our model adheres to physical constraints in hand-object interactions for various grasps of the same object. Fig. 10 showcases grasping poses for objects with diverse geometric shapes from the test set, highlighting our model's ability to generate effective grasps across different objects consistently.

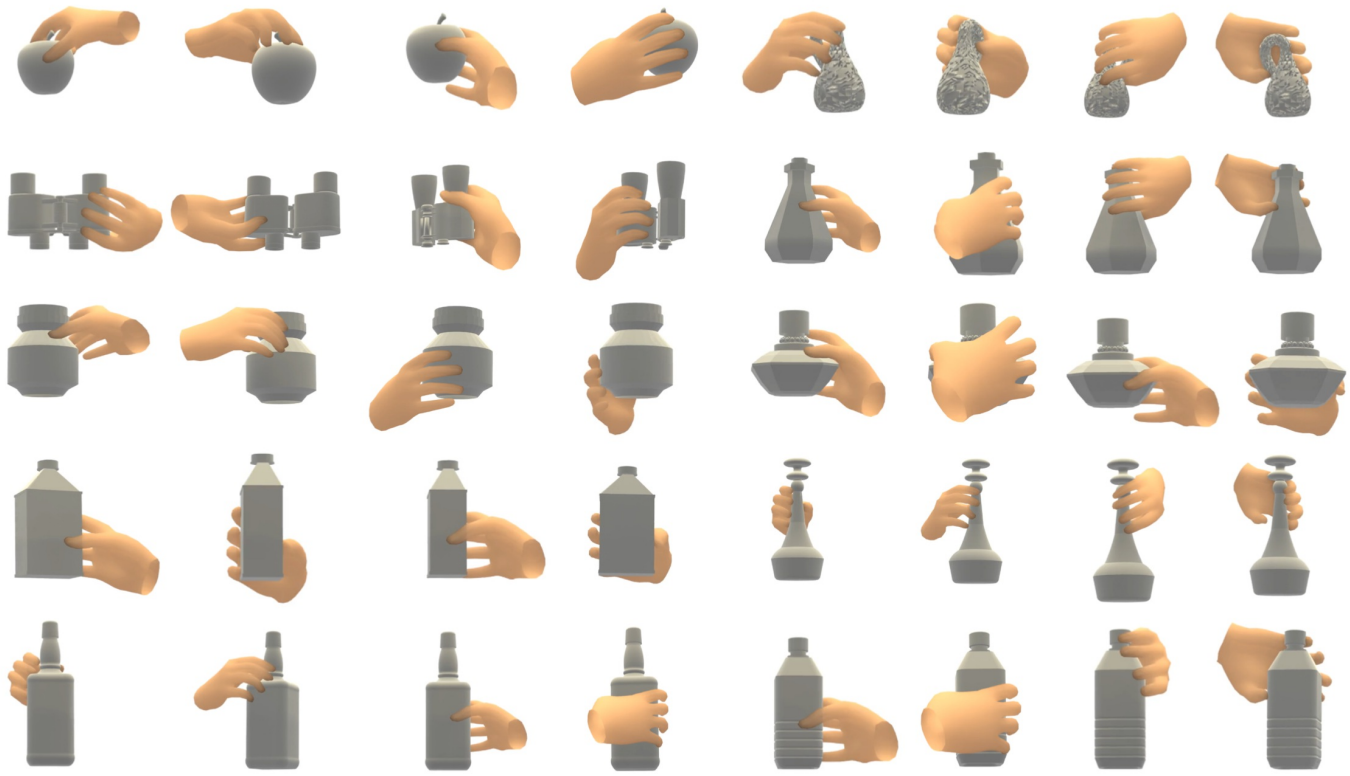

Figure 9. In the visualization results of the autoencoder, we selected two different grasping poses for each object, each shown from two different perspectives.

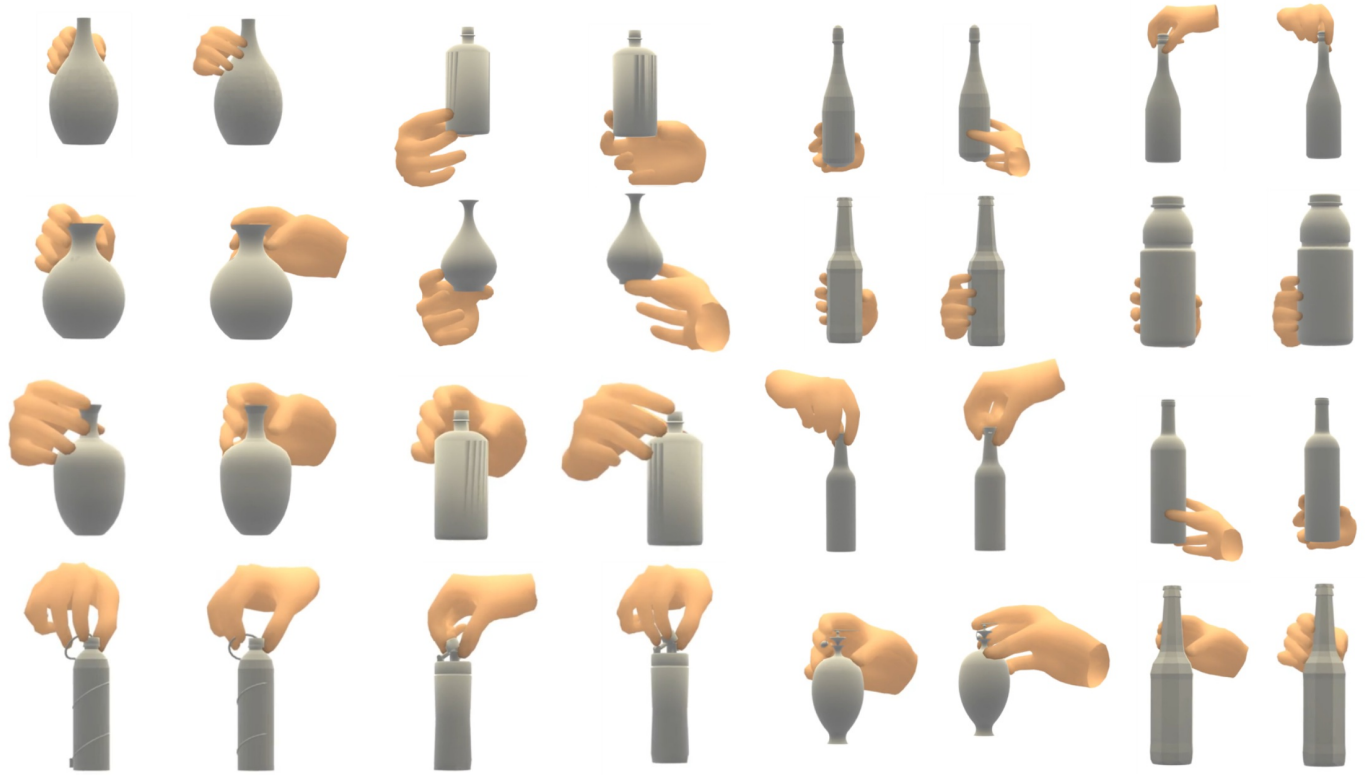

Figure 10. In the autoencoder visualization results, we randomly selected grasping poses, each shown from two different perspectives.

Acknowledgements

This work was supported by NSFC 62350610269, Shanghai Frontiers Science Center of Human- centered Artificial Intelligence, and MoE Key Lab of Intelligent Perception and Human-Machine Collaboration (ShanghaiTech University).

References

1. ^*Tzionas D, Ballan L, Srikantha A, Aponte P, Pollefeys M, Gall J (2015). "Capturing hands in action using discriminative salient points and physics simulation". International Journal of Computer Vision. **118**: 172-193. S2CID 16842481.*
2. ^*Chen Z, Hasson Y, Schmid C, Laptev I (2022). "AlignSDF: Pose-Aligned Signed Distance Fields for Hand-Object Reconstruction". ArXiv. **abs/2207.12909**. S2CID 251067116.*
3. ^*Doosti B, Naha S, Mirbagheri M, Crandall DJ (2020). "HOPE-Net: A Graph-Based Model for Hand-Object Pose Estimation". 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pages 6607-6616. S2CID 214719923.*
4. a, b*Liu SW, Jiang H, Xu J, Liu S, Wang X (2021). "Semi-Supervised 3D Hand-Object Poses Estimation with Interactions in Time". 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pages 14682-14692. S2CID 235377407.*
5. ^*Chen Z, Chen S, Schmid C, Laptev I (2023). "gSDF: Geometry-Driven Signed Distance Functions for 3D Hand-Object Reconstruction". 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pages 12890-12900. S2CID 258298107.*
6. ^*Cha J, Kim J, Yoon JS, Baek S (2024). "Text2HOI: Text-guided 3D Motion Generation for Hand-Object Interaction". ArXiv. **abs/2404.00562**. S2CID 268819822.*
7. ^*Costabile MF, Paternò F (2005). "Human-computer interaction: INTERACT 2005: IFIP TC13 International Conference, Rome, Italy, September 12-16, 2005: proceedings". S2CID 60475063.*
8. ^*Höll M, Oberweger M, Arth C, Lepetit V. Efficient physics-based implementation for realistic hand-object interaction in virtual reality. 2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR) 2018:175-182. S2CID 4106937.*
9. ^*Farulla GA, Pianu D, Cempini M, Cortese M, Russo LO, Indaco M, Nerino R, Chimienti A, Oddo CM, Vitiello N (2016). "Vision-Based Pose Estimation for Robot-Mediated Hand Telerehabilitation". Sensors (Basel, Switzerland). **16**. S2CID 16776545.*
10. a, b, c, d, e, f, g*Karunratanakul K, Yang J, Zhang Y, Black MJ, Muandet K, Tang S. Grasping field: Learning implicit representations for human grasps. In: 2020 International Conference on 3D Vision (3DV); 2020 Nov. doi:10.1109/3dv50981.2020.00043.*
11. a, b, c, d, e, f, g, h, i, j, k, l, m*Karunratanakul K, Spurr A, Fan Z, Hilliges O, Tang S. "A skeleton-driven neural occupancy representation for articulated hands." In: 2021 International Conference on 3D Vision (3DV). IEEE; 2021. p. 11-21.*
12. a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q*Liu S, Zhou Y, Yang J, Gupta S, Wang S (2023). "Contactgen: Generative contact*

modeling for grasp generation". Proceedings of the IEEE/CVF International Conference on Computer Vision 20609--20620.

13. [a, b, c, d, e, f, g, h, i, j, k] *Jiang H, Liu S, Wang J, Wang X. Hand-object contact consistency reasoning for human grasps generation. Proceedings of the IEEE/CVF international conference on computer vision 2021:11107-11116.*
14. [^] *Sohn K, Lee H, Yan X (2015). "Learning Structured Output Representation using Deep Conditional Generative Models". In: Cortes C, Lawrence N, Lee D, Sugiyama M, Garnett R, editors. Advances in Neural Information Processing Systems. Curran Associates, Inc.; 2015.* ***28****. Available from: https://proceedings.neurips.cc/paper_files/paper/2015/file/8d55a249e6baa5c06772297520da2051-Paper.pdf.*
15. [a, b] *Preechakul K, Chatthee N, Wizadwongsa S, Suwajanakorn S (2021). "Diffusion Autoencoders: Toward a Meaningful and Decodable Representation". 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pages 10609-10619. S2CID 244729224.*
16. [a, b, c, d, e, f, g] *Romero J, Tzionas D, Black MJ (2022). "Embodied hands: Modeling and capturing hands and bodies together". arXiv preprint arXiv:2201.02610. arXiv:2201.02610.*
17. [a, b, c, d, e, f, g, h, i, j] *Hampali S, Rad M, Oberweger M, Lepetit V. "HOnnotate: A method for 3D Annotation of Hand and Object Poses". In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2020 Jun. doi:10.1109/cvpr42600.2020.00326.*
18. [a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p] *Yang L, Li K, Zhan X, Wu F, Xu A, Liu L, Lu C (2022). "OakInk: A Large-Scale Knowledge Repository for Understanding Hand-Object Interaction". In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).*
19. [a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y] *Taheri O, Ghorbani N, Black MJ, Tzionas D. GRAB: A dataset of whole-body human grasping of objects. In: Computer Vision--ECCV 2020: 16th European Conference, Glasgow, UK, August 23--28, 2020, Proceedings, Part IV 16. Springer; 2020. p. 581--600.*
20. [^] *Kulkarni N, Rempe D, Genova K, Kundu A, Johnson J, Fouhey DF, Guibas LJ (2023). "NIFTY: Neural Object Interaction Fields for Guided Human Motion Synthesis". ArXiv.* ***abs/2307.07511****. S2CID 259924851.*
21. [a, b, c, d] *Wu Y, Wang J, Zhang Y, Zhang S, Hilliges O, Yu F, Tang S (2022). "Saga: Stochastic whole-body grasping with contact". In: European Conference on Computer Vision. Springer. pp. 257–274.*
22. [^] *Ghosh A, Dabral R, Golyanik V, Theobalt C, Slusallek P (2022). "IMoS: Intent-Driven Full-Body Motion Synthesis for Human-Object Interactions". Computer Graphics Forum.* ***42****. S2CID 254685591.*
23. [^] *Liu M, Tang S, Li Y, Rehg JM (2019). "Forecasting Human Object Interaction: Joint Prediction of Motor Attention and Egocentric Activity". ArXiv.* ***abs/1911.10967****. S2CID 208267647.*
24. [^] *Brahmbhatt S, Ham C, Kemp CC, Hays J. "ContactDB: Analyzing and Predicting Grasp Contact via Thermal Imaging". 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019:8701-8711. S2CID 118643835.*
25. [^] *Brahmbhatt S, Tang C, Twigg CD, Kemp CC, Hays J (2020). "ContactPose: A Dataset of Grasps with Object Contact and Hand Pose". ArXiv.* ***abs/2007.09545****. S2CID 220647075.*
26. [^] *Chao YW, Yang W, Xiang Y, Molchanov P, Handa A, Tremblay J, Narang YS, Van Wyk K, Iqbal U, Birchfield S,*

Kautz J, Fox D (2021). "DexYCB: A Benchmark for Capturing Hand Grasping of Objects". 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pages 9040-9049. S2CID 233210016.

27. ^*Li Y, Fu JL, Pollard NS (2007). "Data-driven grasp synthesis using shape matching and task-based pruning".IEEE Transactions on Visualization and Computer Graphics. **13**: 732–747.*
28. ^*Pollard NS, Zordan VB (2005). "Physically based grasping control from example". In:Symposium on Computer Animation. S2CID 15945304.*
29. ^*Zhang H, Ye Y, Shiratori T, Komura T (2021). "ManipNet".ACM Transactions on Graphics (TOG). **40**: 1–14. S2CID 235176037.*
30. ^*Grady P, Tang C, Twigg CD, Vo M, Brahmbhatt S, Kemp CC (2021). "ContactOpt: Optimizing Contact to Improve Grasps". 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pages 1471–1481. S2CID 233240869.*
31. ^*Jiang H, Liu S, Wang J, Wang X (2021). "Hand-Object Contact Consistency Reasoning for Human Grasps Generation". 2021 IEEE/CVF International Conference on Computer Vision (ICCV). pages 11087-11096. S2CID 233169019.*
32. ^*Kry PG, Pai DK (2005). "Interaction capture and synthesis".ACM SIGGRAPH 2006 Papers. S2CID 13937505.*
33. ^*Li Q, Wang J, Loy CC, Dai B. "Task-Oriented Human-Object Interactions Generation with Implicit Neural Representations". 2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). 2023:3023-3032. S2CID 257687817.*
34. ^*Ye Y, Li X, Gupta A, De Mello S, Birchfield S, Song J, Tulsiani S, Liu S (2023). "Affordance Diffusion: Synthesizing Hand-Object Interactions". 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pages 22479-22489. S2CID 257663466.*
35. ^*Zheng J, Zheng Q, Fang L, Liu Y, Yi L (2023). "CAMS: Canonicalized Manipulation Spaces for Category-Level Functional Hand-Object Manipulation Synthesis". 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pages 585-594. S2CID 257771325.*
36. ^*Zhou K, Bhatnagar BL, Lenssen JE, Pons-Moll G (2022). "TOCH: Spatio-Temporal Object-to-Hand Correspondence for Motion Refinement". European Conference on Computer Vision. S2CID 250919519.*
37. ^*Brahmbhatt S, Handa A, Hays J, Fox D (2019). "ContactGrasp: Functional Multi-finger Grasp Synthesis from Contact". 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pages 2386-2393. S2CID 102352660.*
38. ^*Detry R, Kraft D, Buch AG, Krüger N, Piater JH. "Refining grasp affordance models by experience".2010 IEEE International Conference on Robotics and Automation. 2010:2287-2293. S2CID 7422120.*
39. ^*Hsiao K, Lozano-Perez T (2006). "Imitation Learning of Whole-Body Grasps".2006 IEEE/RSJ International Conference on Intelligent Robots and Systems. pages 5657–5662. S2CID 2468294.*
40. ^*Tekin B, Bogo F, Pollefeys M (2019). "H+O: Unified Egocentric Recognition of 3D Hand-Object Poses and Interactions". 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pages 4506-4515. S2CID 131774180.*

41. ^Hasson Y, Varol G, Tzionas D, Kalevatykh I, Black MJ, Laptev I, Schmid C (2019). "Learning Joint Reconstruction of Hands and Manipulated Objects". 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pages 11799-11808. S2CID 106404030.
42. a, bCorona E, Pumarola A, Aleny\`a G, Moreno-Noguer F, Rogez G. GanHand: Predicting Human Grasp Affordances in Multi-Object Scenes. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2020:5030-5040. S2CID 219962806.
43. a, bLiu Y, Yang Y, Wang Y, Wu X, Wang J, Yao Y, Schwertfeger S, Yang S, Wang W, Yu J, et al. Realdex: Towards human-like grasping for robotic dexterous hand. arXiv preprint arXiv:2402.13853. 2024.
44. a, bHo J, Jain A, Abbeel P (2020). "Denoising diffusion probabilistic models".Advances in neural information processing systems. **33**: 6840–6851.
45. ^Sohl-Dickstein JN, Weiss EA, Maheswaranathan N, Ganguli S (2015). "Deep Unsupervised Learning using Nonequilibrium Thermodynamics". ArXiv. **abs/1503.03585**. S2CID 14888175.
46. ^Liu N, Li S, Du Y, Torralba A, Tenenbaum JB (2022). "Compositional Visual Generation with Composable Diffusion Models". ArXiv. **abs/2206.01714**. S2CID 249375227.
47. ^Poole B, Jain A, Barron JT, Mildenhall B (2022). "DreamFusion: Text-to-3D using 2D Diffusion"ArXiv. **abs/2209.14988**. S2CID 252596091.
48. ^Watson D, Chan W, Martin-Brualla R, Ho J, Tagliasacchi A, Norouzi M (2022). "Novel View Synthesis with Diffusion Models". ArXiv. **abs/2210.04628**. S2CID 252780361.
49. ^Lyu Z, Wang J, An Y, Zhang Y, Lin D, Dai B (2023). "Controllable mesh generation through sparse latent point diffusion models". Proceedings of the IEEE/CVF conference on computer vision and pattern recognition pages 271–280.
50. a, bKwon M, Jeong J, Uh Y (2022). "Diffusion Models already have a Semantic Latent Space".ArXiv. **abs/2210.10960**. S2CID 253018703.
51. a, b, cQi C, Su H, Mo K, Guibas LJ. "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation". 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016: 77-85. S2CID 5115938.
52. ^Rombach R, Blattmann A, Lorenz D, Esser P, Ommer B (2021). "High-Resolution Image Synthesis with Latent Diffusion Models". arXiv. arXiv:2112.10752 [cs.CV].
53. a, bSong J, Meng C, Ermon S. "Denoising Diffusion Implicit Models".arXiv:2010.02502. **Preprint**, October 2020. Available from: https://arxiv.org/abs/2010.02502.