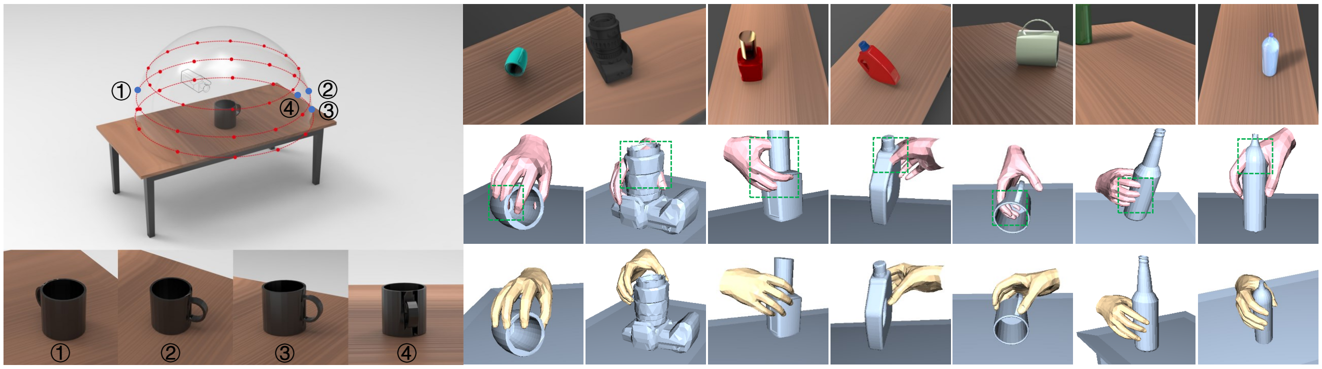
Single-View Scene Point Cloud Human Grasp Generation

Yan-Kang Wang[1] Chengyi Xing[2] Yi-Lin Wei[1] Xiao-Ming Wu[1] Wei-Shi Zheng[1,3*]
[1]School of Computer Science and Engineering, Sun Yat-sen University, China
[2]Stanford University, Stanford, CA
[3]Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, China
{wangyk65, weiylin5, wuxm65}@mail2.sysu.edu.cn chengyix@stanford.edu wszheng@ieee.org



Dataset Scene   Red hands (second row): GraspCVAE   Yellow hands (third row): Ours

Figure 1. The **scene** of our dataset and **comparison** between our model and GraspCVAE [25]. The top-left image depicts the scene in our dataset, comprising a table and an object. Below it are images from four random viewpoints out of 36. The three rows of images on the right side represent, respectively, the single-view images, the generation results of GraspCVAE and our method. The green box indicates the area where the hand-object penetration occurs due to the lack of global perception ability of GraspCVAE.

Abstract

In this work, we explore a novel task of generating human grasps based on single-view scene point clouds, which more accurately mirrors the typical real-world situation of observing objects from a single viewpoint. Due to the incompleteness of object point clouds and the presence of numerous scene points, the generated hand is prone to penetrating into the invisible parts of the object and the model is easily affected by scene points. Thus, we introduce ***S2HGrasp****, a framework composed of two key modules: the Global Perception module that globally perceives partial object point clouds, and the DiffuGrasp module designed to generate high-quality human grasps based on complex inputs that include scene points. Additionally, we introduce* ***S2HGD*** *dataset, which comprises approximately 99,000 single-object single-view scene point clouds of 1,668 unique objects, each annotated with one human grasp. Our extensive experiments demonstrate that S2HGrasp can not only generate natural human grasps regardless of scene points, but also effectively prevent penetration between the hand and invisible parts of the object. Moreover, our model showcases strong generalization capability when applied to unseen objects. Our code and dataset are available at https://github.com/iSEE-Laboratory/S2HGrasp.*

1. Introduction

Learning and capturing hand-object interactions stands as a foundation for understanding human behaviors and has many practical applications. From augmented and virtual reality [24] to the domains of robotic grasps, human-computer interaction [35, 41] and learning from human demonstrations [13, 15, 37, 45], the comprehension and replication of these interactions are of significance.

In this work, we focus on understanding the interactions involved in generating 3D human grasps for objects. We recognize most existing grasp generation methods depend on full 3D object models [25, 26], yet in real-world scenarios, objects and scenes are commonly perceived from a single viewpoint, leading to incomplete object point clouds with potential scene noise. Consequently, existing methods find it challenging to capture the complete geometric fea-

[*] Corresponding author

tures of a partial object and generate human grasps based on such single-view point clouds, limiting their practical applicability. Clearly, a major challenge of this task is the incompleteness of the object. If the model cannot perceive the object's overall shape, the generated grasp is likely to penetrate into the invisible parts of the object. Furthermore, due to interference from scene points, there is a risk that the generated hand will collide with the scene elements.

To address the challenges, we introduce **S2HGrasp**, a novel framework designed for generating human grasps on single-view scene point clouds. S2HGrasp incorporates a Global Perception module that facilitates the global understanding of partial object point clouds, and a DiffuGrasp module aiming at generating high-quality human grasps from complex inputs that contain numerous scene points.

Specifically, the Global Perception module employs a multi-task learning paradigm, including single-view point cloud completion and classification tasks. This module helps to capture the object's global geometric features, and effectively prevents the generated hands from penetrating into the object's invisible parts. During testing, we can directly extract global features from the input single-view point clouds without the need of point cloud completion and classification tasks. Through feature alignment in latent space, our module can capture global features from the partial object in a single-stage process. Concurrently, the DiffuGrasp module is designed to generate human grasps and it adds noise to the normalized hand parameters when training. Then it conditions on the scene features and predicts the original hand parameters. In testing, the DiffuGrasp module progressively denoises a random noise with the condition of scene features, and predicts the final reasonable hand parameters. Leveraging its powerful conditional generation capabilities, the DiffuGrasp module is capable of generating high-quality grasps that are close to the object, even when the input includes scene points.

In addition, we introduce a novel dataset named **S2HGD**, specifically for human grasp generation on single-view scene point clouds. Constructed using the BlenderProc [11] simulator and derived from OakInk dataset [51], S2HGD has a collection of 1,668 unique objects. Each object is individually placed on a tabletop setting. RGB-D images are captured across 36 distinct viewpoints, and corresponding single-view point clouds are derived from depth information. Overall, S2HGD comprises roughly 99,000 point clouds, each paired with a reasonable human grasp.

Our experimental results indicate that our end-to-end approach S2HGrasp outperforms existing generation methods and two-stage methods without data preprocessing and TTA (test-time adaptation, which fails to work well in our task due to the incompleteness of the objects). Additionally, our model not only generates natural and plausible grasps but also exhibits the ability to generalize to unseen objects.

In summary, our main contributions are as follows: 1) We explore a new task of human grasp generation on single-view scene point clouds and achieve favorable results. 2) We propose a novel model named S2HGrasp, which employs a Global Perception module to endow the model with the ability to globally perceive single-view point clouds, and a DiffuGrasp module to generate high-quality grasps despite the complex input that contains the object and scene points. 3) We construct S2HGD, a new dataset for human grasp generation based on single-object single-view scene point clouds. Our code and dataset will both be available.

2. Related Work

Hand-object interaction and grasp. Hand-object interaction is currently receiving increasing attention and involves two main tasks: 1) reconstructing 3D models of hands and objects based on images [6, 12, 16–18, 27, 32, 33, 43, 50] and 2) predicting how to grasp an object based on its visual representation (images or point clouds) [4, 8, 10, 25, 26, 29, 42, 52]. GraspTTA [25] can generate human grasps based on the consistency between hand contact points and object contact regions given complete objects. Affordance Diffusion [52] employs diffusion model to generate hand-object interacting images given object images. When it comes to dexterous hand grasp, [49] and [44] work well using generative models and reinforcement learning, and [29] use a single-shot network to predict dexterous hand grasp in clutter. Existing works mainly focus on generating grasps for complete objects, however, in real world, objects are often observed from a single viewpoint. Thus we study human grasp generation on single-view scene point clouds, employing two modules to tackle the challenges of incompleteness of the object and impact of scene elements.

3D point cloud completion. 3D point cloud completion enables the model to have global perception capability and has various methods, including point-based [30, 47, 54], view-based [21, 55], convolution-based [48], graph-based [34], generative model-based [23], transformer-based [53] approaches, etc. Point Completion Network (PCN) [54] is a learning-based shape completion method, and its decoder designed for point cloud features does not require any assumptions about the shape or category of the point clouds. It can generate both coarse and fine-grained completions with a relatively small number of parameters. Motivated by PCN [54], we design a Global Shape Perception (GSP) to complete partial point clouds. Compared to other SOTA methods for point cloud completion, this approach allows for the presence of scene points and achieves a satisfactory completion result, giving the model global awareness.

Diffusion model. Diffusion models have emerged as the new deep generative models that progressively destruct data by injecting noise, then learn to reverse this process for sam-

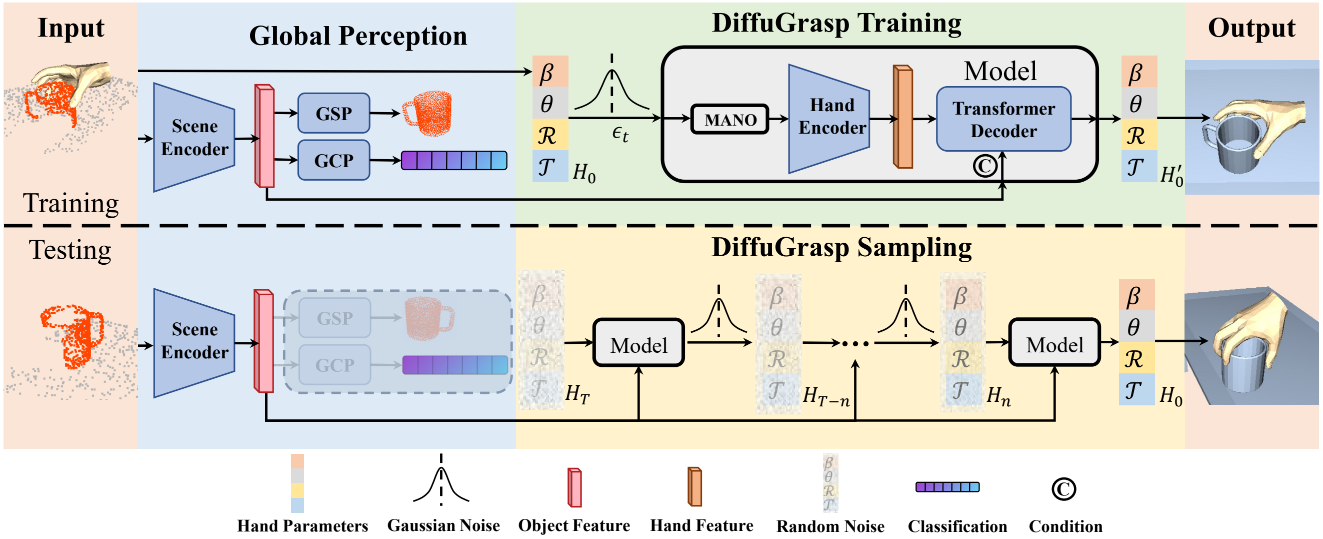


Figure 2. **S2HGrasp framework**. The scene encoder takes single-view scene point clouds as input and extracts their features through PointNet++ [36] and a transformer block. The features are then used in point cloud completion (GSP), classification (GCP), and grasp generation (DiffuGrasp). GSP and GCP won't be used in testing. In the DiffuGrasp Training, the model adds noise to the normalized hand parameters and extracts hand features after passing the parameters into the MANO layer [39]. Then the object features and hand features will be fed into a transformer decoder to predict the original hand parameters. When testing, the DiffuGrasp Sampling starts from a random noise and iteratively denoises it, resulting in final hand parameters.

ple generation. They have been used in a variety of domains, like computer vision [3, 5, 20, 57], natural language processing [1, 28], temporal data modeling [9] and multimodal modeling [2]. Notably, [22] introduces a SceneDiffuser for 3D scene understanding, and it also generates dexterous hands by adding noise to hand parameters and denoising them. Differently, we convert noised hand parameters into point clouds, extract their features, and then decode them to get final grasps. This improves the naturalness of generated hands, as demonstrated later in our experiments.

3. S2HGrasp

3.1. Problem Statement and Method Overview

We explores a novel task: generating physically plausible human grasps for objects based on single-view scene point clouds. We enable the model to globally perceive partial objects and generate high-quality grasps despite interference from scene points. The input is point clouds $\mathcal{P} \in \mathbb{R}^{N\times 3}$ (where N is the number of points including tabletop and partial object), captured from a single viewpoint. The output is human grasp parameters $\mathcal{H} \in \mathbb{R}^{61}$ modeled via differentiable MANO layer [39], with shape parameter $\beta \in \mathbb{R}^{10}$ for size of the hand and pose parameter $\theta \in \mathbb{R}^{45}$ for joint angle of 15 joints, as well as $\mathcal{R} \in \mathbb{R}^3$ and $\mathcal{T} \in \mathbb{R}^3$ for rotation and translation of wrist joint respectively. Given the hand parameters, MANO layer will output the hand mesh with $\hat{\mathcal{M}} = (\hat{\mathcal{V}} \in \mathcal{R}^{778\times 3}, \hat{\mathcal{F}})$, where $\hat{\mathcal{V}}$ and $\hat{\mathcal{F}}$ denote hand vertices and faces. This task is more aligned with real-world scenarios and is beneficial for practical applications.

The generated grasps should be not only natural and plausible, but also able to firmly hold objects in physics-based simulators. Due to the incompleteness of object point clouds, the model must possess the capability to globally perceive the object to prevent hands from penetrating into the invisible parts of the object. Moreover, the model needs to generate high-quality human grasps despite interference from scene points. Therefore, we propose a model named **S2HGrasp**, as summarized in Fig. 2, which consists of two modules: Global Perception module and DiffuGrasp module, to generate human grasps on single-view point clouds.

S2HGrasp is an end-to-end framework. The Global Perception module equips the model with the ability to perceive global geometric features of partial objects, while the DiffuGrasp module is responsible for resisting interference from scene points and generating high-quality human grasps. In the training process, our model takes single-view point clouds and corresponding hand parameters as input and uses an encoder to extract scene features. The Global Perception module endows these features with the global perception of the object. The DiffuGrasp module conditions on these features, adds noise to hand parameters, and trains a decoder to recover the original hand parameters. In testing, DiffuGrasp starts by sampling a random noise and progressively denoises it to derive the final hand parameters.

3.2. Global Perception

The Global Perception module aims at enabling the model to perceive the global geometric characteristics of the object from single-view point clouds. Thanks to the multi-task learning paradigm in training, our model can directly extract the scene features that contain global perception of the partial object in a single-stage process when testing. The resulting scene features will be used in the DiffuGrasp module, making the generated human grasps more natural, avoiding penetration with invisible parts of the object. Our scene point cloud features $\mathcal{F}_s$ are primarily extracted using PointNet++ [36], and we utilize a transformer block similar to [56] to add attention mechanisms, making the model focus more on the local regions suitable for grasping.

Firstly, we design a Global Shape Perception (GSP) to perform point cloud completion task on the extracted feature, motivated by PCN [54]. After getting the feature $\mathcal{F}_s$ of single-view scene point clouds, we pass it through our GSP and result in $\mathcal{N}$ points of the completed object. Subsequently, containing the global perception of the object, $\mathcal{F}_s$ can be better utilized in the human grasp generation.

Secondly, we design a Global Category Perception (GCP) to perform point cloud classification task. Specifically, we feed the obtained scene feature $\mathcal{F}_s$ into a classification head to output the object category. Since the object is incomplete and the input contains both the object and scene points, this classification task is challenging. Nevertheless, we calculate the F1 score for the classification results of S2HGrasp trained on S2HGD, achieving a score of 0.96. It indicates that GCP performs well, enabling the model to focus more on the object. By predicting the object's category, it can better perceive the object's overall shape.

Global Perception Loss. In the GSP of the Global Perception module, we use a Chamfer Distance loss L_{GSP} between the completed point clouds and the corresponding complete object point clouds for supervision, enabling it to have the capability to perceive the global feature of the object. In the GCP, we use Cross Entropy Loss L_{GCP}. The total loss of the Global Perception module is defined as,

$$L_{Glo} = \lambda_{GSP} \cdot L_{GSP} + \lambda_{GCP} \cdot L_{GCP}. \tag{1}$$

3.3. DiffuGrasp

Since the input for our task is single-view scene point clouds including the tabletop and the object, the presence of numerous scene points can affect the quality of the generated hand and may lead to collision with the scene points. Therefore, we design the DiffuGrasp module with two innovations, to generate high-quality and diverse human grasps from complex input. Besides, we also design a plane loss to avoid collision between the hand and the tabletop.

First, we add noise to hand parameters, but direct noise injection is not feasible due to the differences in the ranges and meanings of the parameters. Therefore, we perform specific normalization for each parameter based on its meaning and range. Second, hand parameters include hand shape, pose, and position information, and directly decoding parameters does not effectively model these information. Thus, we pass them through MANO layer [39] to convert them into point clouds and use hand encoder to obtain their features before feeding them into the decoder.

Following the practices of conditional diffusion model [20, 38], in our work, the DiffuGrasp module is with condition $\mathcal{F}_s$, *i.e.*, the feature of single-view point clouds. The DiffuGrasp module can progressively generate human grasps that are close to the object and avoid penetration with the object and the scene during the denoising process.

- Training Process. Conditioning on the single-view scene point cloud feature $\mathcal{F}_s$, the DiffuGrasp module adds noise to the normalized hand parameters $\mathcal{H} \in \mathbb{R}^{61}$. After getting the hand feature $\mathcal{F}_h$, we feed it together with $\mathcal{F}_s$ into a transformer decoder and train the decoder at various noise levels to predict the original hand parameters, *i.e.*, denoise $\mathcal{H}_0$ for different $\mathcal{H}_t$ given t, shown as below:

$$\mathcal{H}_t = \sqrt{1-\overline{\alpha}_t}\mathcal{H}_0 + \sqrt{\overline{\alpha}_t} \cdot \epsilon, \mathcal{H}'_0 = \text{Model}(\mathcal{H}_t), \tag{2}$$

where ϵ is noise, α is noise coefficient and $\overline{\alpha}_t = \prod_{i=1}^{t} \alpha_i$.

- DiffuGrasp Loss. Referring to the codes in DDPM [19], calculating the loss on the input and calculating the loss on the noise are equivalent, so we use *Reconstruction Loss* along with *Cmap Loss*, *Penetration Loss* and *Plane Loss* to train our transformer decoder. Specifically, 1) *Reconstruction Loss* of hand parameters and hand point clouds $\mathcal{P}^h \in \mathcal{R}^{778\times 3}$ is used to minimize reconstruction error,

$$L_{param} = |\mathcal{H}'_0 - \mathcal{H}_0|, L_{\mathcal{V}} = ||\hat{\mathcal{V}} - \mathcal{P}^h||_2^2. \tag{3}$$

2) *Cmap Loss* is used to ensure the generated hand is as close to the object as possible. We compute the distance from the hand's contact points $\mathcal{V}^p$ to the object's point clouds. For each point $\mathcal{P}^o_i$, the distance $\mathbf{D}(\mathcal{P}^o_i) = \min_j ||\mathcal{V}^p_j - \mathcal{P}^o_i||_2^2$. If the distance is less than a threshold, we consider the point as a potentially contactable point on the object, *i.e.*, contact map. One of our training objectives is to make the contact points on the hand as close as possible to the contact map on the object as,

$$L_{cmap} = \sum\nolimits_i \mathbf{D}(\mathcal{P}^o_i), \textit{for all}\ \mathbf{D}(\mathcal{P}^o_i) \le \mathcal{T}. \tag{4}$$

3) *Penetration Loss* is used to penalize cases where the hand penetrates into the object. Specifically, we define the subset of object points that penetrate into the hand's mesh as $\mathcal{P}^o_{in}$, and the penetration loss is defined as minimizing the distances from these points to their nearest points on the hand,

$$L_{penetr} = \frac{1}{|\mathcal{P}^o_{in}|} \sum_{p\in\mathcal{P}^o_{in}} \min_i ||p - \hat{\mathcal{V}}_i||_2^2. \tag{5}$$

Note that we don't use the complete object to calculate the Cmap loss and the Penetration loss, and it is only used in the GSP module. Instead, we combine the input single-view point clouds $\mathcal{P}^s$ with the corresponding completion result $\mathcal{P}^c$ from the GSP for loss calculations. Specifically, we first calculate the minimum distance $\mathbf{D}(\mathcal{P}^c_i) = \min_j ||\mathcal{P}^s_j -$

$\mathcal{P}_i^c||_2^2$, and keep the points $\hat{\mathcal{P}^c} = \{\mathcal{P}_i^c \in \mathcal{P}^c \mid \mathbf{D}(\mathcal{P}_i^c) \geq \theta\}$. We get the final completion result $\mathcal{P}^C = \hat{\mathcal{P}^c} \cup \mathcal{P}^s$, which is used for loss calculation. This approach prevents large errors from directly using rough completion results to compute loss functions. It also allows the model to have a global perception of the object's shape and avoids generating hands that penetrate into the unseen parts of the object. 4) *Plane Loss.* As for generating human grasps based on scene point clouds, one of the challenges is to avoid contact between the hand and the tabletop point clouds. Thus, we design a plane loss to penalize cases where there is penetration between the tabletop point clouds and the hand. To avoid the enormous computational cost of calculating distances between tabletop point clouds and hand vertices, we regress the parameters of the tabletop plane. We define the hand points which are on the backside of the plane as $\hat{\mathcal{V}}_{back}$, and calculate the distance $\mathbf{D}(\hat{\mathcal{V}}_i, P)$ from each point $\hat{\mathcal{V}}_i$ in $\hat{\mathcal{V}}_{back}$ to the plane P, and the plane loss is defined as,

$$L_{plane} = \sum\nolimits_i \mathbf{D}(\hat{\mathcal{V}}_i, P),\ \textit{for all}\ \hat{\mathcal{V}}_i \in \hat{\mathcal{V}}_{back}. \tag{6}$$

Thus, the loss for the DiffuGrasp is:

$$\begin{aligned} L_{Diff} = \lambda_{param} \cdot L_{param} + \lambda_{\mathcal{V}} \cdot L_{\mathcal{V}} + \\ \lambda_c \cdot L_{cmap} + \lambda_{pe} \cdot L_{penetr} + \lambda_{pl} \cdot L_{plane} \end{aligned} \tag{7}$$

- Sampling Process. The Sampling Process begins by randomly sampling a noise from standard Gaussian distribution. Subsequently, it undergoes a denoising process for T timesteps with the feature of single-view point clouds serving as a condition, and ultimately yields the final hand parameters. The normalization follows the same procedure as described in the Training Process. We add noise calculated from $\mathcal{H}_t$ and t to hand parameters $\mathcal{H}_t$, and then feed them into the **Model**, which includes the MANO layer, Hand Encoder, and the Transformer Decoder as shown in Fig. 2, to get $\mathcal{H}_{t-1}$. After T iterations of the denoising process, the model will get final $\mathcal{H}'_0$. The distribution $p_\theta\left(\mathcal{H}_0|\mathcal{F}_s\right)$ represents our sampling process and can be shown as,

$$p_\theta\left(\mathcal{H}_0|\mathcal{F}_s\right) = p\left(\mathcal{H}_T\right) \prod_{t=1}^{T} p\left(\mathcal{H}_{t-1}|\mathcal{H}_t, \mathcal{F}_s\right), \tag{8}$$

$$p(\mathcal{H}_{t-1}|\mathcal{H}_t, \mathcal{F}_s) = \mathcal{N}(\mathcal{H}_{t-1}; \mu_\theta(\mathcal{H}_t, t, \mathcal{F}_s), \sum_\theta(\mathcal{H}_t, t, \mathcal{F}_s)) \tag{9}$$

$$\mathcal{H}_{t-1} = \sqrt{\overline{\alpha}_{t-1}}\mathcal{H}_0 + \sqrt{1 - \overline{\alpha}_{t-1} - \sigma_t^2} \cdot \epsilon_\theta^{(t)}(\mathcal{H}_t) + \sigma_t \epsilon_t, \tag{10}$$

where $\sigma_t = \eta\sqrt{\frac{1-\overline{\alpha}_{t-1}}{1-\overline{\alpha}_t} \cdot (1 - \frac{\overline{\alpha}_t}{\overline{\alpha}_{t-1}})}$, η is DDIM [40] sampling coefficient and $\epsilon_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ is standard Gaussian noise. For more details, please refer to supplementary file.

4. S2HGD Dataset

To better align with real-world scenarios where objects are typically observed from a single viewpoint and to further explore generating human grasps based on single-view scene point clouds, we construct the **S2HGD** dataset. Here we show how we build our S2HGD with BlenderProc [11]:

Data Source. Our raw data (including objects and human grasp annotations) all come from the OakInk dataset [51]. We select a total of 1,667 objects from 35 different categories (*e.g.* bottle, knife, mug) as the dataset's objects.

Simulation Environment Setting. In the simulation environment, we select a table from ShapeNet [7] and position 36 cameras above the table, arranged in three concentric circular arrays at heights of 0.25 meters, 0.45 meters, and 0.65 meters from the table respectively. Each circular array contains 12 cameras at the same height, with a 30-degree spacing between each camera, as shown in Fig. 1.

Data Processing. We drop an object from 20 cm above the table, allowing it to fall onto the tabletop freely. This is repeated ten times to get ten different poses. For each pose, we capture an RGBD image from each camera and convert it into single-view point clouds. After collecting all point clouds, we test all grasps of each object, discard any that contact the table and remove scenes lacking collision-free grasp annotations. From the remaining grasps, we choose the one closest to the object as the final annotation.

To enable our model to generate high-quality grasps for both varying viewpoints of seen objects and entirely unseen objects, we partition our dataset in two unique ways: **View-S2HGD** focuses on varying viewpoints, whereas **Object-S2HGD** is based on object diversity. Specifically, View-S2HGD includes different single-view point clouds of the same objects in training and testing sets. Object-S2HGD ensures that the objects in the testing set are entirely absent from the training set, allowing us to assess the model's capacity for generalization to unseen objects.

In total, S2HGD contains 1,667 objects and about 99,000 single-view scene point clouds with one annotation each.

5. Experiments

5.1. Datasets and Experiment Setup

Implementation Details. The single-view point clouds in our dataset consist of 2,000 points, with 1,000 points belonging to the object and 1,000 points to the tabletop. In training, we use Adam optimizer and LR = $1e-4$, where the LR is reduced by half when the model trained 100, 160 and 180 epochs. We totally train for 200 epochs and the batch size is 60. Before training the entire model with View-S2HGD and Object-S2HGD, we pre-train the GSP module using single-view point clouds and their corresponding complete objects from each dataset respectively for 200 epochs. The loss weights we use are $\lambda_{GSP} = 5$, $\lambda_{GCP} = 2$, $\lambda_{param} = 300$, $\lambda_V = 15$, $\lambda_{cmap} = 300$, $\lambda_{pe} = 15$, $\lambda_{pl} = 1$ for View-S2HGD. For Object-S2HGD, only λ_{pe}

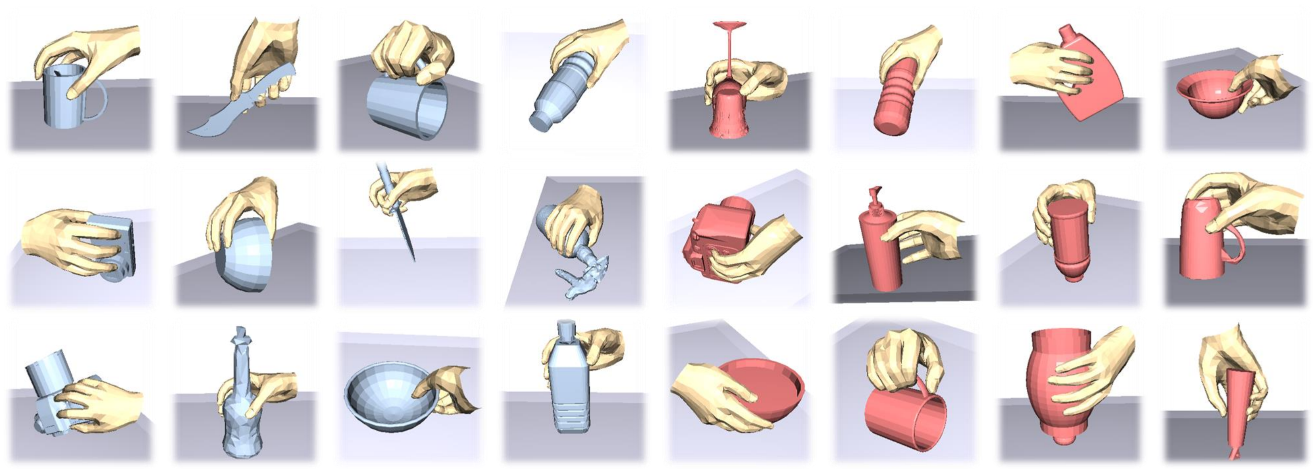


Figure 3. The visualization of generated grasps of our S2HGrasp on our two datasets. The blue objects on the left represent the results of View-S2HGD, while the red objects on the right represent the results of Object-S2HGD.

		View-S2HGD			Object-S2HGD			HO3D-object		Obman-object	
		GT	GC [25]	Ours	GT	GC [25]	Ours	GC [25]	Ours	GC [25]	Ours
Penetration	Depth(cm) ↓	0.27	0.23	**0.21**	0.26	0.25	**0.21**	2.11	**1.38**	0.32	**0.21**
	Volume(cm^3)↓	6.60	13.54	**6.58**	7.22	14.48	**5.58**	8.66	**4.87**	9.84	**4.62**
Grasp Displace	**Mean(cm)↓**	2.01	3.10	**2.73**	2.18	3.69	**3.26**	3.61	**3.30**	4.18	**3.25**
	Variance(cm) ↓	±2.73	±3.49	**±3.16**	±2.69	±3.94	**±3.38**	±3.93	**±3.73**	±4.31	**±3.75**
Perceptual Score	{1,. . . ,5} ↑	4.54	2.60	**3.25**	4.52	2.54	**3.14**	2.11	**2.89**	2.04	**3.13**
Contact	Ratio(%)↑	100	98.07	**99.41**	100	97.87	**98.67**	97.62	**99.06**	85.82	**96.06**
Diversity(σ^2)	Axis(10^{-2}) ↑	2.15	0.85	**1.03**	2.29	0.52	**0.71**	0.50	**0.79**	0.56	**0.70**
	Angle(10^{-2}) ↑	8.86	3.29	**4.51**	9.68	2.04	**2.75**	1.32	**1.95**	2.28	**2.67**

Table 1. Quantitative results compared with GraspCVAE [25] on S2HGD. The objects in HO3D and Obman datasets are used to demonstrate the generalization capability. The **Penetration Volume** metric and the **Grasp displace** metric are our main indicators. The results show that our method outperforms GraspCVAE on all indicators (in bold) and is significantly effective in reducing hand-object penetration caused by incomplete point clouds. The Penetration results of our method are even **better than GT** (with a blue background).

and λ_{param} differ, being set to 20 and 250, respectively.

Datasets. We use S2HGD, Obman [16] and HO3D [14] for our experiments. S2HGD is the dataset we mainly conduct our experiments on. We train and test our model on both View-S2HGD and Object-S2HGD to demonstrate the overall performance. We also conduct ablation experiments on View-S2HGD. Obman and HO3D are used for studying hand-object interaction and Obman is synthetic, while HO3D is real. To validate generalization capability of our method on unseen objects, we utilize the Object-S2HGD for training and testing. We also select 30 objects from Obman and 10 objects from HO3D to construct two testing sets using the same methodology employed in the construction of S2HGD, to further assess generalization capability.

Evaluation Metrics. The primary goal of our model is to generate plausible and natural human grasps for objects based on single-view scene point clouds. Therefore, our main evaluation metrics involve calculating the *penetration* depth and volume between the object and the hand. We also use a physics-based simulation environment to calculate the *grasp displace* to assess the stability of the hand grasping the object. *Perceptual score* is used to evaluate the naturalness of the generated grasps and we invite 50 participants to rate the naturalness of generated grasps, using a scale from 1 to 5, where higher scores indicate higher naturalness. What's more, to demonstrate the contact between the generated hand and the object, we also calculate *contact ratio* between the hand and the object. Besides, to demonstrate diversity of generated grasp poses, we introduce a *diversity* metric. Specifically, we calculate the variance of the rotation axes σ^2_{axis} and angles σ^2_{angle} for 15 joints, excluding the wrist joint, across all grasp samples. Details about the evaluation metrics can be found in the supplementary file.

5.2. Grasp Generation Performance

Visualization results. We visualize the generated grasps on both the View-S2HGD and Object-S2HGD datasets, as shown in Fig. 3. The results demonstrate that our model can generate natural and plausible human grasps for different single-view point clouds of seen objects and shows a notable level of generalization for unseen objects. We also visualize the performance of our GSP module, *i.e.* the partial object point cloud completion results, as shown in the left of Fig. 4. The first row is the input single-view point

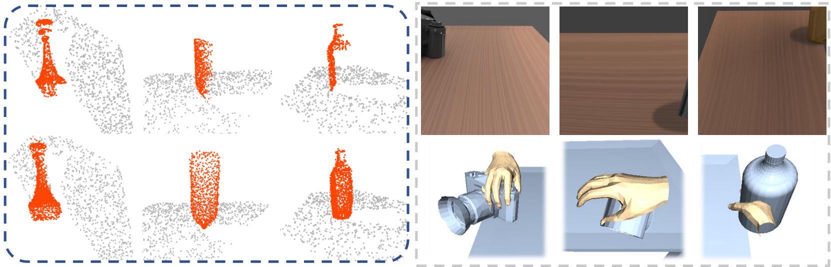


Figure 4. Visualization of point cloud completion and our failure cases. **Left**: Point cloud completion results, with the input single-view point clouds in the top row and completion results below (gray points for the tabletop, red for the object). **Right**: Failure cases, with single-view images at the top and corresponding failure cases of our method below.

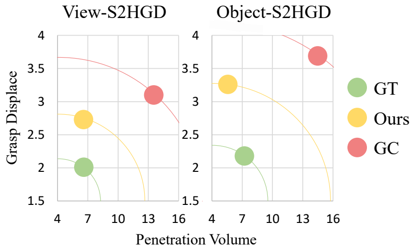


Figure 5. The balance between penetration volume (x-axis) and grasp stability (y-axis). Method whose result point is closer to the origin is more effective.

Method	Penetration		Grasp Displace	Contact
	Depth↓	Volume↓	Mean±Variance↓	Ratio↑
w/o encoder [22]	0.21	6.70	3.13±3.40	98.49
w/ encoder(ours)	0.21	**6.58**	**2.73±3.16**	**99.41**

Table 2. Comparison with method [22] that without hand encoder.

Method	Penetration		Grasp Displace	Contact
	Depth↓	Volume↓	Mean±Variance↓	Ratio↑
PCN+GC	0.22	11.34	6.00±4.83	85.66
PCN+DG	0.20	7.68	6.50±4.51	72.85
Ours	0.21	**5.58**	**3.26±3.38**	**98.67**

Table 3. Comparison with two-stage methods. We use our model trained on Object-S2HGD to compare with others. The two-stage methods initially complete the single-view point cloud (by PCN [54]), and generate human hand grasps on the completed point cloud using GraspCVAE (GC) [25] and our DiffuGrasp(DG).

clouds and the second row is the completion results. The output of our GSP module consists of only the red points. We also visualize some failure cases, shown in the right part of Fig. 4. The first example fails due to the complex shape of the object and it's difficult to perceive its global shape from only partial point clouds, leading to the generated hand penetrating into the object. The last two examples fail because only a small part of the object is visible from the viewpoint. There are too few points to generate a reasonable grasp. More visualization results are in the supplementary file, including the HO3D and Obman datasets.

Comparing with GraspCVAE. As shown in Table 1, we conduct experiments on our View-S2HGD and Object-S2HGD datasets and the testing sets we construct using the objects selected from HO3D [14] and Obman [16] datasets. We compare our model with the SOTA method GraspTTA [25] for the task of generating human grasps based on complete object point clouds. GraspTTA employs ContactNet for Test-Time Adaptation (TTA), which predicts contact areas with the condition of complete object point clouds. Since our task involves single-view scene point clouds as input, with partial object point clouds and interference from scene point clouds, TTA is challenging to apply. Thus, in the comparative experiments, we did not include the TTA module, only utilizing the GraspCVAE network (GC).

The results indicate that our model outperforms GraspCVAE across all datasets. From the "Penetration" metric, it can be seen that GraspCVAE, due to its lack of global perception of single-view point clouds, produces grasps that are significantly penetrate into the object, while ours are much less, even better than GT. Additionally, our model also demonstrates superior generalization capability on unseen objects compared to GraspCVAE. Regarding the "Grasp Displace" metric, a larger penetration volume may result in better grasping stability (the larger the penetration, the greater the force on the fingers in the simulator), which could make the results of GraspCVAE seem favorable. However, an ideal grasp should have both smaller penetration volume and good stability. Therefore, we present the balance on these two metrics of both methods in Fig. 5. It is evident that the points representing our method are closer to the origin point, which also indicates that our method better unifies penetration with grasp stability, meaning the generated grasps are more natural and reasonable.

To generate more natural and stable grasps, instead of directly passing hand parameters into the decoder as done in [22], we first convert the hand parameters into point clouds and extract their features. We then feed the features into the decoder to predict the hand parameters. Our method incorporates more shape and positional information about the hand than direct decoding the hand parameters. To verify the effectiveness of our method, we conduct comparative experiments with the approach without the hand encoder used in [22], as shown in Table 2. The results show that, with a similar penetration volume, our method achieves higher grasp stability (smaller grasp displace), thereby effectively facilitating the generation of higher quality grasps and demonstrating the advantages of our approach.

Comparing with two-stage methods. For the task of generating human grasps based on single-view scene point clouds, one intuitive approach is to first utilize point cloud completion methods to complete the single-view point

Method	CVAE	DiffuGrasp	GP	penetration		Grasp Displace	Contact
				Depth(cm)↓	Volume(cm^3)↓	Mean ± Variance(cm)↓	Ratio(%)↑
CVAE	✓			0.23	13.54	3.10 ± 3.49	98.07
DiffuGrasp		✓		0.20	11.54	3.06 ± 3.33	98.80
DiffuGrasp + GP		✓	✓	**0.21**	**6.58**	**2.73 ± 3.16**	**99.41**

Table 4. Ablation results of the DiffuGrasp module and the Global Perception module(GP) on our View-S2HGD dataset. The role of DiffuGrasp is demonstrated through a comparison with CVAE.

clouds. Subsequently, a generation method like GraspTTA [25] is employed to generate grasps on the completed point clouds. [46] and [31] utilize such two-stage methods to generate dexterous hand grasps in single-view scenes.

We first train a Point Completion Network (PCN) [54] to complete the input single-view point clouds into full objects with tabletop points. Then, we train GraspCVAE (GC) [25] and our DiffuGrasp (DG) to generate human grasps based on these completed results. The testing results are shown in Table 3. The results demonstrate that our method outperforms both two-stage methods by a large margin. The reason is that the two-stage method is not end-to-end, which leads to accumulated errors in the process, impacting the quality of generated grasps. In addition, most works of point cloud completion are not suitable for such a situation with scene points, which will lead to a decrease in the accuracy of point cloud completion. However, the two-stage method has very high requirements for the completion results of single-view point clouds. If the result is poor, the generated grasp can easily penetrate into the object or not make contact with it. In contrast, our end-to-end model effectively avoids such problems. Last, the overall models of two-stage methods tend to be larger, require longer and more tedious training processes, and are less straightforward to train compared to our end-to-end method.

5.3. Ablation Study

We conduct an ablation study on our S2HGrasp using our View-S2HGD dataset to highlight the role of the Global Perception module and demonstrate the advantages of our DiffuGrasp module over CVAE, as shown in Table 4.

Global Perception. The comparative results, as well as the visualizations of point cloud completion, highlight that this module helps the object encoder capture global object characteristics. This, in turn, assists the generation models in producing more natural, collision-free human grasps.

Specifically, by comparing the results of "DiffuGrasp" and "DiffuGrasp+GP", we can find that with the Global Perception module, our model can effectively reduce the volume of hand-object penetration (from 11.54 to 6.58) and enhance the stability of the grasp. This also demonstrates that, with the aid of the Global Perception module, our model has a good perception of the overall shape of local objects, and can effectively prevent the generated hand from penetrating into the invisible areas of the object.

DiffuGrasp. The previous work GraspTTA [25] leverages CVAE to generate human grasps based on the full object model, but suffers from severe model collapse and can't generate diverse grasps poses. We believe that the generative capabilities of CVAE are not sufficient, especially for the more challenging task of generating human grasps based on single-view scene point clouds. Therefore, we conduct comparative experiments on our View-S2HGD dataset using our DiffuGrasp and CVAE (without TTA, as TTA cannot be applied to incomplete point clouds) to demonstrate the advantages of our DiffuGrasp in this task.

Results show that even without the Global Perception module, the penetration volume of DiffuGrasp is smaller than that of CVAE (11.54 to 13.54). Moreover, DiffuGrasp achieves higher grasp stability than CVAE with smaller penetration, which also shows that the grasps generated by DiffuGrasp are more reasonable and natural. In short, despite the incompleteness of the object and the interference from numerous scene points, DiffuGrasp outperforms CVAE and can generate stable human grasps with minimal penetration into the object. Additional ablation studies and experiments are shown in the supplementary file.

6. Conclusion

In this work, we explore a new task of generating human grasps based on single-view scene point clouds rather than full object models. We propose S2HGrasp, along with a new synthetic dataset S2HGD, to address the problem of hand-object penetration caused by the incompleteness of object point clouds. We design a Global Perception module to globally perceive partial objects and a DiffuGrasp module to generate plausible and natural human grasps despite numerous scene points. Our S2HGrasp effectively overcomes the limitation of existing approaches and extends human grasp generation beyond full object models, benefiting the study of hand-object interactions. The experimental results demonstrate that S2HGrasp outperforms other methods and achieves satisfying performance and generalization capabilities across different datasets and unseen objects.

Acknowledgments. This work was supported partially by the National Key Research and Development Program of China(2023YFA1008503),NSFC(U21A20471,U1911401), Guangdong NSF Project (No. 2023B1515040025, 2020B1 515120085).

References

[1] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 2021. 3

[2] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 3

[3] Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. *arXiv preprint arXiv:2112.03126*, 2021. 3

[4] Samarth Brahmbhatt, Ankur Handa, James Hays, and Dieter Fox. Contactgrasp: Functional multi-finger grasp synthesis from contact. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019. 2

[5] Ruojin Cai, Guandao Yang, Hadar Averbuch-Elor, Zekun Hao, Serge Belongie, Noah Snavely, and Bharath Hariharan. Learning gradient fields for shape generation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, 2020. 3

[6] Zhe Cao, Ilija Radosavovic, Angjoo Kanazawa, and Jitendra Malik. Reconstructing hand-object interactions in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 2

[7] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 5

[8] Jiayi Chen, Mi Yan, Jiazhao Zhang, Yinzhen Xu, Xiaolong Li, Yijia Weng, Li Yi, Shuran Song, and He Wang. Tracking and reconstructing hand object interactions from point cloud sequences in the wild. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023. 2

[9] Nanxin Chen, Yu Zhang, Heiga Zen, Ron J Weiss, Mohammad Norouzi, and William Chan. Wavegrad: Estimating gradients for waveform generation. *arXiv preprint arXiv:2009.00713*, 2020. 3

[10] Enric Corona, Albert Pumarola, Guillem Alenya, Francesc Moreno-Noguer, and Grégory Rogez. Ganhand: Predicting human grasp affordances in multi-object scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020. 2

[11] Maximilian Denninger, Martin Sundermeyer, Dominik Winkelbauer, Youssef Zidan, Dmitry Olefir, Mohamad Elbadrawy, Ahsan Lodhi, and Harinandan Katam. Blenderproc. *arXiv preprint arXiv:1911.01911*, 2019. 2, 5

[12] Bardia Doosti, Shujon Naha, Majid Mirbagheri, and David J Crandall. Hope-net: A graph-based model for hand-object pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020. 2

[13] Guillermo Garcia-Hernando, Edward Johns, and Tae-Kyun Kim. Physics-based dexterous manipulations with estimated hand poses and residual reinforcement learning. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020. 1

[14] Shreyas Hampali, Mahdi Rad, Markus Oberweger, and Vincent Lepetit. Honnotate: A method for 3d annotation of hand and object poses. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020. 6, 7

[15] Ankur Handa, Karl Van Wyk, Wei Yang, Jacky Liang, Yu-Wei Chao, Qian Wan, Stan Birchfield, Nathan Ratliff, and Dieter Fox. Dexpilot: Vision-based teleoperation of dexterous robotic hand-arm system. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020. 1

[16] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019. 2, 6, 7

[17] Yana Hasson, Bugra Tekin, Federica Bogo, Ivan Laptev, Marc Pollefeys, and Cordelia Schmid. Leveraging photometric consistency over time for sparsely supervised hand-object reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[18] Yana Hasson, Gül Varol, Cordelia Schmid, and Ivan Laptev. Towards unconstrained joint hand-object reconstruction from rgb videos. In *2021 International Conference on 3D Vision (3DV)*, 2021. 2

[19] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 2020. 4

[20] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *The Journal of Machine Learning Research*, 2022. 3, 4

[21] Tao Hu, Zhizhong Han, and Matthias Zwicker. 3d shape completion with multi-view consistent inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020. 2

[22] Siyuan Huang, Zan Wang, Puhao Li, Baoxiong Jia, Tengyu Liu, Yixin Zhu, Wei Liang, and Song-Chun Zhu. Diffusion-based generation, optimization, and planning in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 3, 7

[23] Zitian Huang, Yikuan Yu, Jiawen Xu, Feng Ni, and Xinyi Le. Pf-net: Point fractal network for 3d point cloud completion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020. 2

[24] Wolfgang Hürst and Casper Van Wezel. Gesture-based interaction via finger tracking for mobile augmented reality. *Multimedia Tools and Applications*, 2013. 1

[25] Hanwen Jiang, Shaowei Liu, Jiashun Wang, and Xiaolong Wang. Hand-object contact consistency reasoning for human grasps generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 1, 2, 6, 7, 8

[26] Korrawe Karunratanakul, Jinlong Yang, Yan Zhang, Michael J Black, Krikamol Muandet, and Siyu Tang. Grasping field: Learning implicit representations for human grasps. In *2020 International Conference on 3D Vision (3DV)*, 2020. 1, 2

[27] Mia Kokic, Danica Kragic, and Jeannette Bohg. Learning to estimate pose and shape of hand-held objects from rgb images. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019. 2

[28] Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *Advances in Neural Information Processing Systems*, 2022. 3

[29] Yiming Li, Wei Wei, Daheng Li, Peng Wang, Wanyi Li, and Jun Zhong. Hgc-net: Deep anthropomorphic hand grasping in clutter. In *2022 International Conference on Robotics and Automation (ICRA)*, 2022. 2

[30] Minghua Liu, Lu Sheng, Sheng Yang, Jing Shao, and Shi-Min Hu. Morphing and sampling network for dense point cloud completion. In *Proceedings of the AAAI conference on artificial intelligence*, 2020. 2

[31] Jens Lundell, Francesco Verdoja, and Ville Kyrki. Ddgc: Generative deep dexterous grasping in clutter. *IEEE Robotics and Automation Letters*, 2021. 8

[32] Meysam Madadi, Sergio Escalera, Xavier Baró, and Jordi Gonzalez. End-to-end global to local cnn learning for hand pose recovery in depth data. *arXiv preprint arXiv:1705.09606*, 2017. 2

[33] Markus Oberweger, Paul Wohlhart, and Vincent Lepetit. Hands deep in deep learning for hand pose estimation. *arXiv preprint arXiv:1502.06807*, 2015. 2

[34] Liang Pan. Ecg: Edge-aware point cloud completion with graph convolution. *IEEE Robotics and Automation Letters*, 2020. 2

[35] Thammathip Piumsomboon, Adrian Clark, Mark Billinghurst, and Andy Cockburn. User-defined gestures for augmented reality. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*. 2013. 1

[36] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 2017. 3, 4

[37] Yuzhe Qin, Yueh-Hua Wu, Shaowei Liu, Hanwen Jiang, Ruihan Yang, Yang Fu, and Xiaolong Wang. Dexmv: Imitation learning for dexterous manipulation from human videos. In *European Conference on Computer Vision*, 2022. 1

[38] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022. 4

[39] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *arXiv preprint arXiv:2201.02610*, 2022. 3, 4

[40] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 5

[41] Srinath Sridhar, Anna Maria Feit, Christian Theobalt, and Antti Oulasvirta. Investigating the dexterity of multi-finger input for mid-air text entry. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 2015. 1

[42] Omid Taheri, Nima Ghorbani, Michael J Black, and Dimitrios Tzionas. 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*, 2020. 2

[43] Bugra Tekin, Federica Bogo, and Marc Pollefeys. H+ o: Unified egocentric recognition of 3d hand-object poses and interactions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019. 2

[44] Weikang Wan, Haoran Geng, Yun Liu, Zikang Shan, Yaodong Yang, Li Yi, and He Wang. Unidexgrasp++: Improving dexterous grasping policy learning via geometry-aware curriculum and iterative generalist-specialist learning. *arXiv preprint arXiv:2304.00464*, 2023. 2

[45] An-Lan Wang, Kun-Yu Lin, Jia-Run Du, Jingke Meng, and Wei-Shi Zheng. Event-guided procedure planning from instructional videos with text supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 1

[46] Wei Wei, Daheng Li, Peng Wang, Yiming Li, Wanyi Li, Yongkang Luo, and Jun Zhong. Dvgg: Deep variational grasp generation for dextrous manipulation. *IEEE Robotics and Automation Letters*, 2022. 8

[47] Xin Wen, Tianyang Li, Zhizhong Han, and Yu-Shen Liu. Point cloud completion by skip-attention network with hierarchical folding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020. 2

[48] Haozhe Xie, Hongxun Yao, Shangchen Zhou, Jiageng Mao, Shengping Zhang, and Wenxiu Sun. Grnet: Gridding residual network for dense point cloud completion. In *European Conference on Computer Vision*, 2020. 2

[49] Yinzhen Xu, Weikang Wan, Jialiang Zhang, Haoran Liu, Zikang Shan, Hao Shen, Ruicheng Wang, Haoran Geng, Yijia Weng, Jiayi Chen, et al. Unidexgrasp: Universal robotic dexterous grasping via learning diverse proposal generation and goal-conditioned policy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 2

[50] Lixin Yang, Xinyu Zhan, Kailin Li, Wenqiang Xu, Jiefeng Li, and Cewu Lu. Cpf: Learning a contact potential field to model the hand-object interaction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 2

[51] Lixin Yang, Kailin Li, Xinyu Zhan, Fei Wu, Anran Xu, Liu Liu, and Cewu Lu. Oakink: A large-scale knowledge repository for understanding hand-object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 2, 5

[52] Yufei Ye, Xueting Li, Abhinav Gupta, Shalini De Mello, Stan Birchfield, Jiaming Song, Shubham Tulsiani, and Sifei

Liu. Affordance diffusion: Synthesizing hand-object interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 2

[53] Xumin Yu, Yongming Rao, Ziyi Wang, Zuyan Liu, Jiwen Lu, and Jie Zhou. Pointr: Diverse point cloud completion with geometry-aware transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2021. 2

[54] Wentao Yuan, Tejas Khot, David Held, Christoph Mertz, and Martial Hebert. Pcn: Point completion network. In *2018 international conference on 3D vision (3DV)*, 2018. 2, 4, 7, 8

[55] Xuancheng Zhang, Yutong Feng, Siqi Li, Changqing Zou, Hai Wan, Xibin Zhao, Yandong Guo, and Yue Gao. View-guided point cloud completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 2

[56] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2021. 4

[57] Dian Zheng, Xiao-Ming Wu, Zuhao Liu, Jingke Meng, and Wei-shi Zheng. Diffuvolume: Diffusion model for volume based stereo matching. *arXiv preprint arXiv:2308.15989*, 2023. 3