






# SEMGRASP: Semantic Grasp Generation via Language Aligned Discretization

Kailin Li<sup>1,2</sup>, Jingbo Wang<sup>2</sup>, Lixin Yang<sup>1</sup>, Cewu Lu<sup>1,2</sup><sup>✉</sup>, and Bo Dai<sup>2</sup>

<sup>1</sup> Shanghai Jiao Tong University, Shanghai, China

{kailinli,siriusyang,lucewu}@sjtu.edu.cn

<sup>2</sup> Shanghai AI Laboratory, Shanghai, China

{wangjingbo,daibo}@pjlab.org.cn

**Abstract.** Generating natural human grasps necessitates consideration of not just object geometry but also semantic information. Solely depending on object shape for grasp generation confines the applications of prior methods in downstream tasks. This paper presents a novel semantic-based grasp generation method, termed **SEMGRASP**, which generates a static human grasp pose by incorporating semantic information into the grasp representation. We introduce a discrete representation that aligns the grasp space with semantic space, enabling the generation of grasp postures in accordance with language instructions. A Multimodal Large Language Model (MLLM) is subsequently fine-tuned, integrating object, grasp, and language within a unified semantic space. To facilitate the training of **SEMGRASP**, we compile a large-scale, grasp-text-aligned dataset named **CAPGRASP**, featuring over 300k detailed captions and 50k diverse grasps. Experimental findings demonstrate that **SEMGRASP** efficiently generates natural human grasps in alignment with linguistic intentions. Our code, models, and dataset are available publicly at: <https://kailinli.github.io/SemGrasp>.

**Keywords:** Semantic Grasp Generation · Discrete representation · MLLM

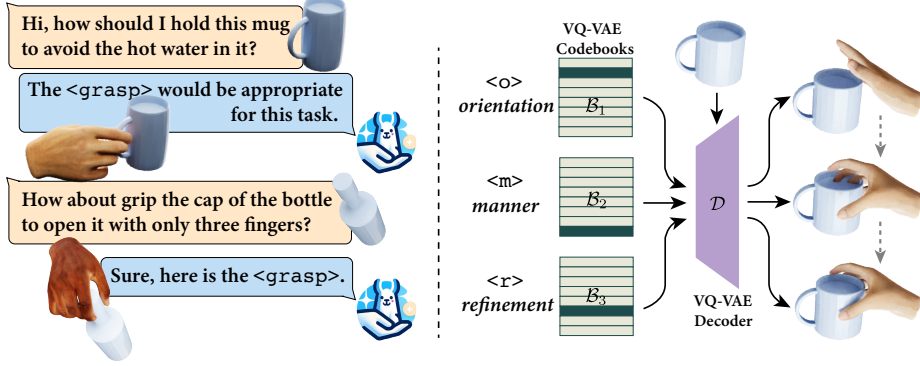
## 1 Introduction

In applications such as AR/VR and embodied robotics, the ability to generate human-like grasps for a given object is of substantial value. The goal of grasping extends beyond simple object lifting; it involves alignment with human intent and preparation for subsequent manipulation tasks, such as avoiding hot water in a mug or preparing to open a bottle cap (Fig. 1 left). Hence, relying solely on the geometric information of objects is inadequate. Combining the semantic information of the object with the description of intent enables the generation of more natural and logical grasps.

Typical grasp representations in previous grasp generation methods exhibit constraints in embedding semantic information. For example, methods that depict grasps through robotic hand poses [31, 44, 47, 50, 70, 74, 78, 93], MANO

---

<sup>✉</sup> Cewu Lu is the corresponding author.



**Fig. 1:** Our **SEMGRASP** methodology. The left figure shows the grasp generation process, while the right illustrates the decoding of our discrete grasp representation. **<grasp>** comprises three tokens: **<o>**, **<m>**, and **<r>**.

model [63] based poses [6, 7, 23, 28, 30, 46, 68, 69, 72, 75, 80, 89], contact regions [3, 20, 37, 38, 45, 81, 82, 87], and implicit forms [32, 33]. Attempts to generate dexterous grasps based on semantic cues, such as those by [28, 47, 78, 81, 93], rely on coarse affordance vectors for conditional generation or directly use vision-language models [18, 60] to filter sampled grasps. Nevertheless, integrating detailed semantic information or language descriptions into the grasp generation process remains challenging.

When humans plan a grasping posture, they initially determine the grasp’s general *orientation*, guided by the object’s category and the semantics of the instruction. Subsequently, the specific *manner* of grasping is decided, influenced by both the manipulation intent and the object’s shape. Finally, the *refinement* of the grasp pose is conducted, taking into account the object’s detailed geometry and the hand-object contact state to ensure physical plausibility. Therefore, it is crucial to design a grasp representation that explicitly incorporates these three steps while implicitly embedding semantic information.

In this paper, we introduce a novel grasp generation method, termed **SEMGRASP**, that incorporates semantic information into the generation process. This approach, inspired by human grasp planning, divides the grasp representation into three interrelated components: 1) *orientation*, influenced by the intent and the object’s function; 2) *manner*, specifying the grasp taxonomy required for interaction; and 3) *refinement*, detailing the hand pose adjustments necessary for physical plausibility.

Given that language is inherently discrete, we naturally extend this discreteness to other modalities to synchronize with the semantic space, similar to other multimodal studies [29, 41, 83]. Furthermore, human grasp poses can be categorized into 33 discrete types according to the Grasp Taxonomy [14]. By combining object shape and manipulation intent, a human grasp can be derived by adjusting these foundational types. Consequently, we employ a vector quantized variational autoencoder (VQ-VAE) [62, 73] to discretize the grasp components

into tokens. This approach not only augments the alignment of grasp-language within the semantic space but also offers significant benefits: 1) it enhances grasp generation’s controllability and interpretability through explicit token representation; 2) it markedly reduces the dimensionality of the grasp space, simplifying the learning process of this representation. The VQ-VAE operates as an encoder-decoder structure, taking the object’s point cloud as conditional input, whereby the encoder encodes the grasp into three tokens via a codebook lookup, and the decoder reconstructs these tokens back into the original grasp.

To align the discrete grasp representation of **SEMGRASP** with semantic space more effectively, we leverage a multimodal large language model (MLLM). Inspired by the structure of LLaVA [43], our MLLM processes three modalities: the discretized grasp tokens, the object features obtained via PointBERT [84], and the language description. Following many MLLM works [29, 43], our model is trained in two training stages: 1) multimodal alignment, training the MLLM to predict grasp tokens from object features and language descriptions, thus mapping these modalities in a unified space; and 2) instruction tuning, fine-tuning the MLLM to enhance grasp generation for complex outputs.

Currently, well-aligned language-grasp datasets are scarce. The annotations in [28, 81] cover only simple intentions. To train **SEMGRASP**, we collect **CAPGRASP**, extending existing hand-object interaction datasets in three ways: 1) Low-level annotations, identifying contact states, such as which fingers touch the object and which parts of the object are being grasped. These details are deduced from the positions of hand and object. 2) High-level annotations, encompass manipulation intent and grasp force, for instance, ‘tightly touch the bottle cap to unscrew it’. Utilizing low-level information, we generate these grasp-related descriptions with GPT-4 [57]. Additionally, based on dataset images or rendered visuals, we employ GPT-4v [58] to annotate complex intentions. 3) Conversational annotations: Employing LLMs, we construct grasp-language mixed conversations. These dialogues include both low and high-level information. We train **SEMGRASP** using **CAPGRASP**. Experiments show that our method well generate the corresponding grasp pose across multiple metrics. We also verify our method’s potential value in AR/VR and embodied robotics applications.

In summary, our contributions are threefold: 1) We propose **SEMGRASP**, an innovative grasp generation method integrating semantic information. 2) We introduce a novel grasp discrete representation, efficiently and effectively expressing grasp postures and supporting the task of grasp description through language alignment. 3) We compile **CAPGRASP** dataset. To the best of our knowledge, it is the first dataset of semantic grasps that encompass low-level, high-level, and conversational annotations.

## 2 Related Works

**Grasp Generation** Grasp generation remains a fundamental task with wide applications in robotics [1, 55]. Recently, the generation of human-like grasps has attracted increasing attention. Unlike the 6DoF (degrees of freedom) parallel-

jaw grippers commonly used in robotics [10, 12, 13, 31, 54, 70], the higher freedom in human fingers significantly complicates grasp generation. ObMan [23] leverages GraspIt! [53] for synthesizing grasps, while [46] optimize grasp based on force closure. Methods such as [6, 74, 78] validate their techniques in physical simulations. Data-driven approaches [6, 7, 23, 28, 30, 46, 48, 68, 69, 72, 75, 82, 89] employ end-to-end generative models like cVAE [67] or GAN [19]. Given the high degrees of freedom in the MANO model [63], most of these works implement post-processing to enhance contact consistency and physical plausibility. However, these methods solely emphasize geometric information of objects or incorporate basic intent features [28, 81]. In contrast, our **SEMGRASP** integrates semantic aspects of grasping, facilitating the direct generation of grasp postures that correspond with language descriptions.

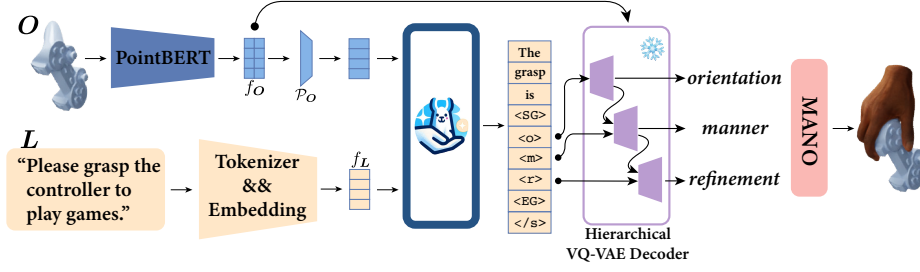
**Hand-Object Interaction Datasets** Understanding hand-object interaction is pivotal in AR/VR, animation, and embodied AI. Existing datasets primarily concentrate on hand-object pose estimation or reconstruction. These datasets are either synthesized through rendering techniques [7, 16, 23, 40] or compiled by annotating real-world data [2, 4, 5, 11, 17, 21, 22, 34, 36, 39, 48, 49, 59, 65, 69, 76, 81, 85, 94]. Some studies delve into the semantics of grasping. For instance, [93] introduces ‘touch codes’ to depict contact states between fingers and object parts. [81] offers annotations of grasping intent and object segmentation based on affordances. [28] segments object point clouds according to grasping semantics. Nonetheless, these datasets typically provide only basic semantic categorizations. Our **CAPGRASP**, in contrast, delivers comprehensive annotations of hand-object interaction, encompassing detailed low-level contact state information, high-level grasp-related descriptions, and conversational annotations.

**Multimodal Large Language Models** The development and application of Large Language Models (LLMs) have surged in recent years. Leading commercial models like GPT-4 [57] and open-source counterparts such as Llama [71] and Vicuna [90] demonstrate exceptional language understanding and generation capabilities. There’s an increasing trend of integrating LLMs into multimodal tasks, including image [26, 43, 92], video [41, 86], 3D [25, 77, 83], and human posture and motion tasks [15, 29]. For example, [43] uses a vision encoder to extract image features and aligns these with the language space using projection layers. [29] interprets motion sequences as a series of tokens, and fine-tunes the T5 model [61] with LoRA [24] to facilitate various tasks like motion generation and captioning. Our work fine-tunes a Vicuna-based model [90] for grasp generation, leveraging LLMs to interpret and generate complex hand-object interactions.

## 3 Method

### 3.1 Overview

Given a specific object  $\mathbf{O}$ , represented as a point cloud, our goal is to align the human grasp  $\mathbf{G}$  with the associated language description  $\mathbf{L}$ , facilitating the task of semantic grasp generation. To this end, we introduce a novel grasp generation methodology, termed **SEMGRASP**, which fundamentally comprises two



**Fig. 2:** Our **SEMGRASP** pipeline. The grasp-aware language model outputs both the grasp tokens and the language conversations.

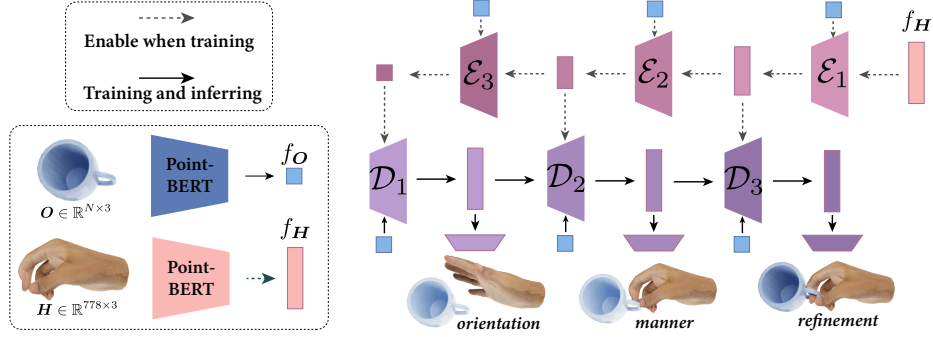
principal components: grasp discretization (Sec. 3.2) and a grasp-aware language model (Sec. 3.3). The initial phase involves the tokenization of the grasp into three interrelated tokens by training a VQ-VAE [62, 73]. As shown in Fig. 1 right, after the tokenizer is trained, the VQ-VAE is frozen, enabling the projection of  $\langle \text{grasp} \rangle$  tokens from the codebook onto grasp configurations. Subsequently, as illustrated in Fig. 2, the grasp-aware language model is designed to reconcile the discrete grasp representations with the linguistic domain, trained specifically to generate corresponding  $\langle \text{grasp} \rangle$  tokens. These resultant  $\langle \text{grasp} \rangle$  tokens can then be reverted to the original grasp pose through the VQ-VAE decoder. The training of the grasp-aware language model is conducted utilizing our dataset, **CAPGRASP**, which builds upon existing dataset of hand-object interactions [81], augmented through automated expansions (Sec. 3.4).

### 3.2 Grasp Discretization

Consistent with prior studies [30, 33], we define the grasp  $\mathbf{G} = (\mathbf{T}, \boldsymbol{\theta}, \boldsymbol{\beta})$  within the canonical space of the object. Here,  $\mathbf{T} \in \mathbb{R}^{4 \times 4}$  represents the homogeneous transformation matrix, indicating the global rotation and translation of the hand relative to the object’s central coordinate system. The parameters  $\boldsymbol{\theta} \in \mathbb{R}^{15 \times 3}$  and  $\boldsymbol{\beta} \in \mathbb{R}^{10}$  denote the local hand pose and shape parameters, respectively. The hand vertices  $\mathbf{H} \in \mathbb{R}^{778 \times 3}$  are computed using a differentiable layer, specifically the MANO  $\mathcal{M}$  model [63], where  $\mathbf{H} = \mathcal{M}(\mathbf{G}) = \mathcal{M}(\mathbf{T}, \boldsymbol{\theta}, \boldsymbol{\beta})$ .

In this work, to more effectively illustrate the human grasp process and align it with the semantic space, we discretize the grasp  $\mathbf{G}$  into three components  $\langle \mathbf{o}, \mathbf{m}, \mathbf{r} \rangle$ , representing the *orientation*, *manner*, and *refinement* token, respectively, where  $\mathbf{o}, \mathbf{m}, \mathbf{r} \in \mathbb{N}$ . We employ a hierarchical VQ-VAE [62], encompassing the trainable codebooks  $\mathcal{B}_i$ , encoders  $\mathcal{E}_i$ , and decoders  $\mathcal{D}_i$ , where  $i \in \{1, 2, 3\}$ , to quantize the grasp vector into meaningful integers and subsequently reconstruct the original grasp vector from the quantized tokens (Fig. 3).

The encoders progressively map the hand’s representation into the latent space, capturing grasp information from low to high levels. This structured approach enables the simulation of the grasping process through conditional probabilities: 1) The hand’s global information  $\mathbf{T}$  is captured with the *orientation*



**Fig. 3:** Simplified illustration of the discretization process with hierarchical VQ-VAE.

token  $\langle o \rangle$ , where  $\hat{T} = \mathcal{D}_1(o, O)$ . 2) The local hand pose  $\theta, \beta$  is encapsulated by the *manner* token  $\langle m \rangle$ , conditioned on  $\langle o \rangle$ , with  $\hat{\theta}, \hat{\beta} = \mathcal{D}_2(o, m, O)$ . 3) For the fine-tuning process, the delta parameters  $\Delta T, \Delta \theta, \Delta \beta$  are represented by the *refinement* token  $\langle r \rangle$ , conditioned on  $\langle o, m \rangle$ , where  $\Delta \hat{T}, \Delta \hat{\theta}, \Delta \hat{\beta} = \mathcal{D}_3(o, m, r, O)$ . The final grasp is reconstructed as  $\hat{G} = (\Delta \hat{T} \cdot \hat{T}, \Delta \hat{\theta} + \hat{\theta}, \Delta \hat{\beta} + \hat{\beta})$ . The hat symbol  $\hat{\cdot}$  denotes the reconstructed values.

Specifically, the codebook  $\mathcal{B} = \{b_k\}_{k=1}^K$ , with each  $b_k \in \mathbb{R}^{d_B}$ , where  $d_B$  represents the dimension of the VQ-VAE latent space and  $K$  the number of codebook entries. For each input vector  $z$ , the encoder  $\mathcal{E}$  maps it to the latent space using the mapping network  $\mathcal{N}_{\mathcal{E}}$  and finds the nearest codebook entry  $b_z$ :  $z = \mathcal{E}(z) = \arg\min_k \|\mathcal{N}_{\mathcal{E}}(z) - b_k\|_2$ , where  $z \in \{1, 2, \dots, K\}$ . The decoder  $\mathcal{D}$  then reconstructs the original vector  $z$  with the mapping network  $\mathcal{N}_{\mathcal{D}}$ :  $\hat{z} = \mathcal{D}(z) = \mathcal{N}_{\mathcal{D}}(b_z)$ . The VQ-VAE is trained to minimize the reconstruction loss  $\mathcal{L}_{\text{rec}}$ , the embedding loss  $\mathcal{L}_{\text{emb}}$ , and the commitment loss  $\mathcal{L}_{\text{com}}$ :

$$\mathcal{L}_{\text{rec}} = \|H - \hat{H}\|_2^2 = \|H - \mathcal{M}(\hat{G})\|_2^2 \quad (1)$$

$$\mathcal{L}_{\text{emb}} + \mathcal{L}_{\text{com}} = \|\text{sg}[\mathcal{N}_{\mathcal{E}}(z)] - b_z\|_2^2 + \|\mathcal{N}_{\mathcal{E}}(z) - \text{sg}[b_z]\|_2^2 \quad (2)$$

where  $\text{sg}[\cdot]$  denotes the stop-gradient operation. To express semantics through  $\langle o, m, r \rangle$ ,  $\mathcal{L}_{\text{rec}}$  comprises three components, detailed in the Appx.

### 3.3 Grasp Aware Language Model

Building upon the grasp discrete representation, we design a grasp-aware language model aimed at facilitating semantic grasp generation tasks. As depicted in Fig. 4, our model is trained to align three distinct modalities: the human grasp  $G$ , object points  $O$ , and the language description  $L$ .

**Grasp Modal** After we train the VQ-VAE, this module is frozen. We take the encoders  $\mathcal{E}$  as the grasp tokenizer that transfers the grasp  $G$  into the  $\langle \text{grasp} \rangle$  token which contains three components:  $\langle o, m, r \rangle$ . To distinct the grasp token from the language, we add special tokens  $\langle \text{SG} \rangle$  and  $\langle \text{EG} \rangle$  to the start and end

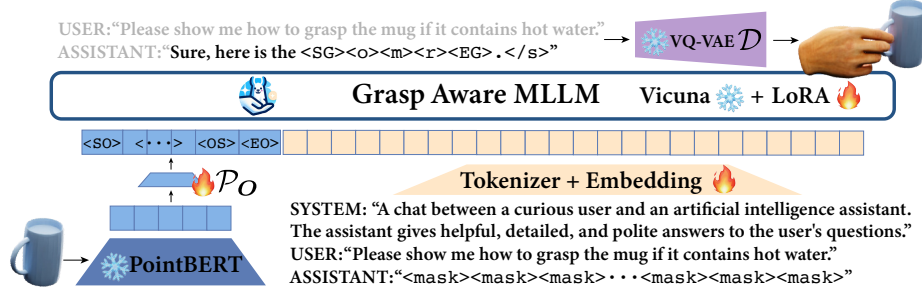


Fig. 4: Grasp Aware Language Model.

of the grasp, respectively. The grasp token, as generated by the MLLM model, can be converted back to the human grasp utilizing the VQ-VAE decoders  $\mathcal{D}$ .

**Object Modal** We employ PointBERT [84] to extract the object features  $f_O$  from the point cloud  $O$ , where  $O \in \mathbb{R}^{N \times 3}$  and  $f_O \in \mathbb{R}^{M \times d_O}$ . Here,  $N$  denotes the number of points,  $M$  the count of point features, and  $d_O$  the dimension of the object feature space. Notably, PointBERT requires the normalization of object sizes, a factor crucial for grasp generation. Thus, we incorporate the object size as a distinct token <OS> within the MLLM inputs. Subsequently, object features are projected into a unified space alongside the grasp token via a linear projection layer  $\mathcal{P}_O$ . Similar to the grasp token, special tokens <S0> and <EO> are affixed to the object feature sequence, delineating its start and end. The projected object features serve as the system message in the GPT-style prompt.

**Language Modal** Our language model undergoes fine-tuning based on the Vicuna-7B checkpoint [90], with a tailored prompt designed to direct the completion of specified tasks (detailed further in Appx). Textual content is tokenized into 32K word pieces employing the SentencePiece methodology [35].

**Training** Our model architecture, akin to LLaVA [43], aligns all modalities through projection or embedding layers into a unified semantic space, yielding  $X = \{x^i\}_{i=1}^T \subset \mathbb{R}_{d_L}^T$ , where  $d_L$  signifies the semantic space dimensionality. An autoregressive language model is then trained to optimize the likelihood of the next tokens, predicated on preceding ones  $p(\hat{X}|X) = \prod_i p(\hat{x}^i | \hat{x}^{<i}, x)$ , terminating with the end-of-sentence token </s>. The primary objective is to diminish the negative log-likelihood loss  $\mathcal{L}_{\text{NLL}}$ :

$$\mathcal{L}_{\text{NLL}} = -\log p(\hat{X}|X) = -\sum_i \log p(\hat{x}^i | \hat{x}^{<i}, x) \quad (3)$$

Utilizing LoRA [24], we finetune the MLLM model, adjusting approximately 0.4B parameters within the linear layers. This process unfolds in two phases: 1) multimodal alignment, wherein the MLLM is trained to predict the grasp tokens based on object features and language descriptions, consolidating these modalities. During this phase, the object feature projection layer  $\mathcal{P}_O$  is updated. The embedding layer of the MLLM is also refined to accommodate the newly introduced special tokens; and 2) instruction tuning, throughout which the MLLM

undergoes further refinement for grasp generation task combined with language outputs, keeping the projection layer frozen for training stability.

### 3.4 CAPGRASP Dataset

Currently, no dataset with well-aligned grasp language annotations exists for training **SEMGRASP** on grasp generation tasks and downstream applications. Due to the high cost and labor-intensive nature of manual semantic annotation, we design an automatic annotation methodology based on GPT-4 to augment existing hand-object interaction datasets. Our dataset, **CAPGRASP**, encompasses low-level, high-level, and conversational annotations.

**Low-level Annotations** Low-level annotations refer to the contact relationships between each finger and various parts of the object. According to the Grasp Taxonomy [14], we can deduce the grasp type and intent from these low-level annotations. For instance, if the thumb and index finger are in contact with a screw, it is inferred that the grasp type is a ‘pinch’ and the intent is to ‘screw/unscrew’. The OakInk dataset [81] provides annotations for objects’ CAD models, hand vertices, and object part segmentation. Utilizing this information, we calculate the contact states when the distance between hand vertices and the object’s part segmentation points is less than an empirically defined threshold.

**High-level Annotations** High-level intent is annotated from two perspectives: 1) based on low-level contact information. Given this information (i.e., the finger and object part contact), we employ GPT-4 to infer the grasping intent. For example, if all fingers are grasping the handle of a mug, GPT-4 can deduce that it is a firm grasp with possible intents such as ‘make a toast’ or ‘avoid hot beverage’. 2) Based on images or rendered views. Since the OakInk dataset includes a subset of real captured images (i.e., OakInk-image), we manually select representative frames that are clear, unobstructed, and with explicit intent. We leverage GPT-4v, a commercial visual-language model, to infer high-level information such as manipulation intent and grasp force. For grasps in OakInk without matching images (i.e., OakInk-shape), we render scenes with Blender using realistic hand textures [42]. The prompt details are elaborated in Appx.

**Conversational Annotations** With the aforementioned low-level and high-level annotations, we construct conversations using the GPT-4 model. We ask GPT-4 to generate various conversational templates from different perspectives, including detailed hand-object contact information, manipulation intent, grasp force, and type. These dialogues must be consistent with the grasp and ensure logical plausibility. Detailed prompts are provided in Appx.

Considering the hallucination problem of GPT-4, to ensure the quality of our **CAPGRASP**, we manually review these annotations to filter out intents that defy common sense and conversations that lack logical coherence. Statistically, our dataset includes approximately 1.8k object models from OakInk, about 50,000 hand-object grasp pairs. For each pair, we offer on average 6 detailed captions and conversational annotations. Additionally, to enhance the performance of our method, we further augment **CAPGRASP** by generating over 1.5M conversational annotations using Llama-3 [52] in a fully automated process.



## 4 Metrics and Experiments

Our methodology, **SEMGRASP**, incorporates two principal components: the grasp discrete representation and the grasp-aware language model. We assess the reconstruction accuracy of VQ-VAE to demonstrate the validity of our grasp discretization approach. Additionally, we evaluate the performance of grasp generation by our MLLM. Comparative and ablation studies underscore the effectiveness of our methodology.

### 4.1 Metrics

**Aspect of Physical Plausibility** To evaluate the physical plausibility of the predicted grasp pose  $\hat{\mathbf{G}}$ , we employ several metrics: 1) Mean Per-Vertex Position Error (MPVPE, in  $mm$ ) calculates the average  $L2$  distance per vertex between the predicted hand mesh  $\hat{\mathbf{H}}$  and the ground truth  $\mathbf{H}$ , when available. 2) Penetration Depth (PD, in  $cm$ ) measures the maximum penetration depth of hand vertices into the object, indicating surface penetration. 3) Solid Intersection Volume (SIV, in  $cm^3$ ) quantifies the volumetric intersection by voxelizing the object mesh and calculating the volume within the hand surface. 4) Simulation Displacement (SD, in  $cm$ ) tests grasp stability in PyBullet [8], measuring the object’s center displacement under steady hand conditions and gravity [23]. These metrics gauge both the quality of grasp generation and the accuracy of our grasp discrete representation.

**Aspect of Semantic Consistency** Semantic consistency is evaluated by examining the quality of grasp generation: 1) GPT-4 assisted evaluation. For generated grasps  $\hat{\mathbf{G}}$ , we first render the hand-object interaction following the same pipeline as in Sec. 3.4. Then, we use GPT-4v to score the semantic consistency of the grasp images based on input captions. Scores range from 0 to 100, with higher scores indicating better consistency. The prompts used in GPT-4 assisted evaluation are listed in Appx. 2) P-FID calculates the Fréchet Inception Distance between the point clouds of the  $\hat{\mathbf{H}}$  and  $\mathbf{H}$ , using the pre-trained feature extractor from [56]. 3) Perceptual Score (PS) assesses the naturalness of grasps and semantic consistency, with 5 volunteers rating the generated grasps on a 5-point Likert scale. The final score is the mean Likert score.

### 4.2 Implementation Details

The VQ-VAE’s codebook  $\mathcal{B}$  consists of  $K = 512$  entries, each dimensioned at  $d_{\mathcal{B}} = 256$ , initialized by sampling uniformly from  $[-\frac{1}{K}, \frac{1}{K}]$ . We employ PointBERT [84] as the point cloud feature extractor in the VQ-VAE encoder  $\mathcal{E}$  for both hand vertices  $\mathbf{H}$  and object vertices  $\mathbf{O}$ . Similar to [77], PointBERT is pretrained using the ULIP-2 method [79] for enhanced geometry-language alignment. The predicted rotation from the VQ-VAE decoders  $\mathcal{D}$  uses a 6D representation [91], subsequently converted to the axis-angle representation for further computation.

For the MLLM, we utilize Llama structure [71] as the model backbone, fine-tuning based on the Vicuna-7B checkpoint [90]. To extract object feature  $f_{\mathbf{O}}$ ,

we reuse PointBERT from the VQ-VAE and freeze its parameters.  $f_{\mathcal{O}}$  includes  $M = 513$  embeddings, each of dimension  $d_{\mathcal{O}} = 384$ . The object projection layer  $\mathcal{P}_{\mathcal{O}}$  projects  $f_{\mathcal{O}}$  to the language space dimension of 4096. We configure the LoRA module [24] with a rank setting of  $r = 64$ , resulting in approximately 6% parameters being finetuned. The learning rate is set to  $5e-4$  and  $3e-5$  for the multimodal alignment stage and instruction tuning stage, respectively, with a cosine annealing learning rate scheduler for training stability. The batch size is 128, and the MLLM is trained over 20 epochs on 4 A100 GPUs with 80GB of memory each.

### 4.3 Comparisons

**Discrete VQ-VAE Grasp Representation** Given the discretization of grasps, three primary concerns arise: 1) Does discretization compromise reconstruction accuracy? 2) Does it affect the physical plausibility of the interaction?, and 3) Does it have the capability to embed semantic information?

To answer the first two questions, we compare our method with two state-of-the-art methods, GrabNet [69] and Jiang *et al.* [30], on a reconstruction task. Both methods are based on cVAEs [67] for grasp generation. For a fair comparison, we retrain these models using our **CAPGRASP** dataset. GrabNet employs RefineNet to refine the interaction in an end-to-end iterative manner, whereas Jiang *et al.* utilize test time adaptation (TTA) to optimize hand-object contact. Our method leverages a *refinement* token to adjust the hand pose in an end-to-end manner. Compared to configurations without the *refinement* token, our approach with the *refinement* token exhibits superior performance, achieving a 26% improvement in MPVPE and a 9% improvement in SIV. Considering the TTA, an optimization-based approach, can precisely improve hand-object interaction, we also report our method’s performance with TTA in Tab. 1 for a fair comparison, which attains current SOTA results in PD and SIV. The results demonstrate that our discrete grasp representation method can accurately depict hand poses and specifically ensure the physical plausibility of interactions. Compared to previous SOTA methods, our method is competitive and exhibits advantages in certain metrics.

**Table 1:** Our discrete VQ-VAE grasp representation compared with SOTA methods.

	MPVPE ↓	PD ↓	SIV ↓	SD <i>mean.</i> ↓	SD <i>std.</i> ↓
<b>CAPGRASP</b> dataset	-	0.11	0.62	0.94	1.62
GrabNet [69] <i>w/o refineNet</i>	18.14	0.76	5.42	1.75	2.61
GrabNet [69]	27.49	0.54	3.45	1.77	2.36
GrabNet [69] <i>w/ TTA</i>	27.16	0.49	2.16	<b>1.35</b>	1.56
Jiang <i>et al.</i> [30] <i>w/o TTA</i>	33.68	0.72	5.81	1.53	1.77
Jiang <i>et al.</i> [30] <i>w/ TTA</i>	33.84	0.58	2.78	1.36	<b>1.55</b>
Ours <i>w/o refinement token</i>	20.36	0.48	3.00	1.95	2.11
Ours	<b>14.97</b>	0.46	2.72	2.14	2.37
Ours <i>w/ TTA</i>	23.61	<b>0.37</b>	<b>1.27</b>	1.90	2.12

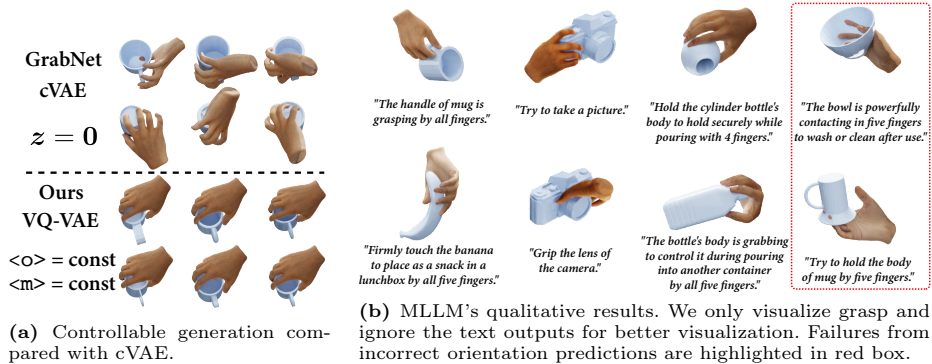
Fig. 5: Qualitative results of our **SEMGRASP**.

Table 2: Quantitative results of our MLLM-based grasp generation method.

	P-FID ↓	PD ↓	SIV ↓	SD <i>mean.</i> ↓	SD <i>std.</i> ↓	GPT-4 ↑	PS ↑
<b>CAPGRASP</b>	-	0.11	0.62	0.94	1.62	82.3	4.7
BERT [9] based	3.32	0.49	4.60	2.17	<b>2.26</b>	47.3	3.7
<b>SEMGRASP</b>	<b>2.28</b>	<b>0.48</b>	<b>4.24</b>	<b>2.00</b>	2.33	<b>74.5</b>	<b>4.6</b>

To explore whether our discrete representation method can encode semantic information, we conduct a controllable generation task. For GrabNet, the cVAE-based method, we directly fix the sampling vector  $z = \mathbf{0}$ , enforcing the model to generate similar grasps. For our method, we assign the same const value to the  $\langle o, m \rangle$  tokens. As depicted in Fig. 5a, take the specific category of mugs as an example, although the shape of the mug varies, our generated grasps maintain consistency in orientation and manner, showing the semantic consistency of our method. In contrast, GrabNet’s predictions lack interpretability.

**Language Guided Grasp Generation** We need to validate that MLLM can control grasp generation  $\mathbf{G}$  based on textual input  $\mathbf{L}$ . To the best of our knowledge, there are no existing works directly comparable to ours. Therefore, leveraging our discrete representation, we construct a straightforward baseline that treats this task as a classification problem. We finetune the official BERT model [9] to embed the language description in conjunction with the object feature. Subsequently, we deploy three distinct classification heads to predict the  $\langle o, m, r \rangle$  tokens. We train this modified BERT with our **CAPGRASP** following the same settings in **SEMGRASP**. These predicted tokens are then decoded into the final grasp pose as in our **SEMGRASP**. The outcomes of this experimental setup are documented in Tab. 2. From the results, we observe that our MLLM outperforms the baseline in both the physical plausibility and semantic consistency metrics. Showing that simply treating the task as a classification problem is not sufficient to generate grasp pose that aligns with the language description. On one hand, benefitted from the pretrained LLM, our **SEMGRASP** can well understand the language instructions. On the other hand, the discrete represen-

**Table 3:** Representation ablation.

	MPVPE ↓	PD ↓	SIV ↓	SD	
				mean. ↓	std. ↓
One token	29.95	0.66	5.14	1.90	2.29
$\langle \mathbf{o}, \mathbf{m} \rangle$	25.73	0.58	4.32	2.14	2.50
$\langle \mathbf{o}, \mathbf{m}, \mathbf{r} \times 2 \rangle$	15.37	0.50	2.98	2.28	2.57
$\langle \mathbf{o}, \mathbf{m}, \mathbf{r} \times 3 \rangle$	15.90	0.52	3.37	1.98	2.18
Single VQ	28.02	0.68	5.31	1.81	2.00
<i>w/o</i> semantic	21.94	0.60	4.59	1.84	1.91

**Table 4:** VQ-VAE settings.

	MPVPE ↓	PD ↓	SIV ↓	SD	
				mean. ↓	std. ↓
$\mathcal{B}$ entries $K = 256$	95.52	1.59	51.83	1.42	1.29
$\mathcal{B}$ entries $K = 1024$	39.82	0.80	5.30	3.10	3.40
$\mathcal{B}$ dim. $d_{\mathcal{B}} = 512$	46.32	1.11	10.86	2.00	3.09
$\mathcal{B}$ dim. $d_{\mathcal{B}} = 128$	38.67	0.80	6.42	2.28	2.78
Vanilla	48.48	1.09	9.42	2.24	2.87
EMA+Reset [62]	24.67	0.73	5.74	1.81	2.00

tation is interpretable, making the MLLM more controllable. We demonstrate the qualitative results of our **SEMGRASP** in Fig. 5b.

#### 4.4 Ablation Studies

**Ablation on Discrete Representation** We conduct ablation studies to examine the design of our tokenization approach. As outlined in Sec. 3.2, we utilize three tokens—*orientation*, *manner*, and *refinement*—to represent  $\langle \mathbf{grasp} \rangle$  as  $\langle \mathbf{o}, \mathbf{m}, \mathbf{r} \rangle$ . In our evaluation, detailed in Tab. 3, we explore different token configurations: 1) Single Token: Compressing  $\mathbf{G}$  into a single codebook significantly degrades reconstruction accuracy. 2) Two Tokens: This setup differs from the *w/o refinement* token setting in Tab. 1, as here we train the representation with only two tokens from scratch. 3) and 4) Multiple *refinement* Tokens: Iteratively predicting  $\langle \mathbf{r} \rangle$  and adjusting the hand pose step by step demonstrates that performance deteriorates when the number of  $\langle \mathbf{r} \rangle$  exceeds one. Moreover, we empirically find that predicting more tokens increases the complexity of MLLM training. 5) Single VQ-VAE: We train a single VQ-VAE to predict three grasp tokens simultaneously with a shared codebook. A single network struggles to encapsulate the intricate grasp representation. 6) *w/o* semantic. In this setting, we still use the hierarchical VQ-VAE to predict three tokens, but we do not assign semantic meaning to the tokens. This setup results in decreased performance.

**Ablation on VQ-VAE settings** Our investigation into the configurations of VQ-VAE focuses on two aspects: 1) Codebook  $\mathcal{B}$  Setting: The size of trainable parameters in the codebook has a significant impact on network performance, leading to either non-convergence or underfitting (see Tab. 4). 2) Training Strategy: VQ-VAE often suffers from codebook collapse. While methods like exponential moving average (EMA) and codebook reset (Reset) are used to mitigate this (as in [62, 88]), we find that these strategies weaken the representation effectiveness in the grasp representation task. Thus, we opt not to use the EMA strategy and allow each entry to be reset only once during training.

**Ablation on MLLM settings** We conduct ablation studies on the MLLM configurations as presented in Tab. 5: 1) Pretrained LLM: The comparison between Llama-7B [71] and Vicuna-7B models shows Vicuna-7B as more aligned with our grasp generation needs, offering better task suitability. 2) Object size

**Table 5:** Ablation study of our MLLM-based grasp generation.

	P-FID ↓	PD ↓	SIV ↓	GPT-4 ↑	PS ↑
w/ Llama	2.38	0.51	4.20	58.9	3.8
w/o <S0>	3.74	0.70	8.20	43.3	3.2
w/o 2-stage	4.54	0.49	5.26	62.5	4.0
LoRA $r = 16$	3.76	0.48	4.38	69.2	3.3
LoRA $r = 128$	2.68	0.50	5.10	74.5	4.0

token <S0>. Contrary to the original ULIP’s size normalization of the object point cloud, our findings highlight the significance of object size for grasp generation tasks. 3) Training Stages: Comparing the MLLM trained in a two-stage process with a single-stage approach shows that the former not only enhances effectiveness but also stabilizes the training process. 4) and 5) LoRA rank: We explore the impact of the LoRA rank on the MLLM. We experimentally find that rank 64 is the optimal choice for our generation task.

## 5 Applications

To demonstrate the real-world applicability of **SEMGRASP**, we conducted case studies in AR/VR and robotics, showing that our method, combined with RL-based policies, can synthesize dynamic grasp motions.

### 5.1 Application in AR/VR

In the context of AR/VR, producing grasps that align with user intent and facilitate natural object manipulation is essential. We evaluated the practicality of grasps generated by **SEMGRASP** using the D-grasp method [6] within the RaiSim [27] simulated environment. D-grasp, which is based on a reinforcement learning (RL) approach, focuses on creating dynamic human grasps. It calculates the next grasp action from a static reference grasp  $\bar{\mathbf{G}}$ , the target object position  $\bar{\mathbf{T}}_{\mathbf{O}}$ , and the current state, including the hand and object’s pose and velocity, using a policy  $\pi$  trained with the PPO [64] algorithm.

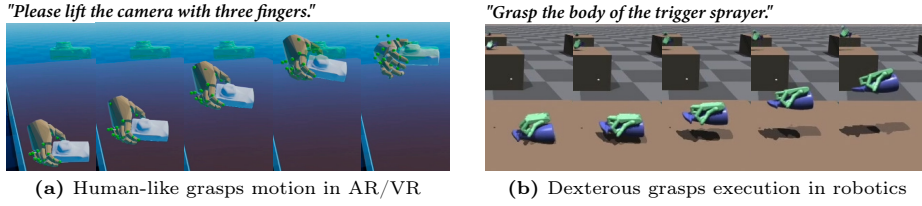
For our experiments, **SEMGRASP** generates the reference pose  $\bar{\mathbf{G}}$  for a specified object  $\mathbf{O}$  and language instruction  $\mathbf{L}$ . We utilize the publicly available D-grasp checkpoints<sup>1</sup> to synthesize dynamic grasps. The object is targeted to lift 10 *cm* upwards along the gravitational direction. We maintain the hand shape parameter  $\beta$  at  $\mathbf{0}$ , consistent with D-grasp. This process is illustrated in Fig. 6a.

### 5.2 Application in Embodied Robotics

Our method’s efficacy is further validated in embodied robotics. Following the UniDexGrasp<sup>2</sup> methodology [78], which entails static reference grasp generation

<sup>1</sup> <https://github.com/christsa/dgrasp>

<sup>2</sup> <https://github.com/PKU-EPIC/UniDexGrasp>



**Fig. 6:** Applications of our **SEMGRASP** in both the AR/VR environment and robotics scenario.

and goal-conditioned grasp execution, our experiments focus on the latter phase to assess the performance of **SEMGRASP**-generated grasps. We retarget our generated grasp  $\bar{G}$  to the ShadowHand  $\mathcal{S}$  [66] and evaluate the grasp within the IsaacGym simulation [51].

Initially, **SEMGRASP** produces the reference grasp pose  $\bar{G}$ . We then establish a pipeline to adapt  $\bar{G}$  for the ShadowHand model  $\bar{G}_{\mathcal{S}}$ , beginning with aligning finger keypoints between the MANO and ShadowHand models. Given the significant differences in degrees of freedom (DoF) and morphology between the two models, we implement a fitting-based optimization approach to refine the ShadowHand-object interaction. After optimizing  $\bar{G}_{\mathcal{S}}$ , we apply the off-the-shelf UniDexGrasp’s pretrained policy to execute the dynamic grasp sequence. Details on our fitting pipeline and the integration of  $\bar{G}_{\mathcal{S}}$  with UniDexGrasp are further elaborated in the Appx. Fig. 6b showcases the process of grasping and lifting as generated by UniDexGrasp.

## 6 Conclusion

We introduce **SEMGRASP**, an approach aimed at generating semantic grasps from language instructions. We propose a novel grasp representation that emulates the natural human grasping process. The discretized representation is both interpretable and controllable, making it ideal for semantic space alignment. Leveraging this representation, we deploy MLLM to generate grasps from language instructions. Tailored for this task, we also present **CAPGRASP**, a comprehensive dataset containing grasp-text-aligned annotations. As we explore potential applications in AR/VR and embodied robotics, we are hopeful that **SEMGRASP** will contribute to advancements in generating more human-like, semantically coherent grasps in various contexts.

**Limitations** Despite **SEMGRASP** demonstrating the capability to generate static single-hand grasps from semantic cues and dynamic grasps through RL integration, exploration remains in two directions: two-hand manipulation and end-to-end semantic grasp motion synthesis. The former requires addressing both hands’ cooperation, and the latter, the continuity of motion, both contingent on the availability of extensive, high-quality motion capture or synthesis data for training. Tackling these challenges promises to advance embodied grasping, pushing toward more sophisticated and realistic manipulation.

## Acknowledgements

This work is supported by the National Key Research and Development Project of China (No. 2022ZD0160102 and No. 2021ZD0110704), as well as the Shanghai Artificial Intelligence Laboratory, XPLOER PRIZE grants.

Cewu Lu, the corresponding author, is a member of Qing Yuan Research Institute, MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University, and the Shanghai AI Laboratory, Shanghai, China.

## References

1. Bohg, J., Morales, A., Asfour, T., Kragic, D.: Data-driven grasp synthesis—a survey. *IEEE Transactions on robotics* (2013)
2. Brahmbhatt, S., Ham, C., Kemp, C.C., Hays, J.: ContactDB: Analyzing and predicting grasp contact via thermal imaging. In: *CVPR* (2019)
3. Brahmbhatt, S., Handa, A., Hays, J., Fox, D.: Contactgrasp: Functional multi-finger grasp synthesis from contact. In: *IROS* (2019)
4. Brahmbhatt, S., Tang, C., Twigg, C.D., Kemp, C.C., Hays, J.: ContactPose: A dataset of grasps with object contact and hand pose. In: *ECCV* (2020)
5. Chao, Y.W., Yang, W., Xiang, Y., Molchanov, P., Handa, A., Tremblay, J., Narang, Y.S., Van Wyk, K., Iqbal, U., Birchfield, S., et al.: Dexycb: A benchmark for capturing hand grasping of objects. In: *CVPR* (2021)
6. Christen, S., Kocabas, M., Aksan, E., Hwangbo, J., Song, J., Hilliges, O.: D-grasp: Physically plausible dynamic grasp synthesis for hand-object interactions. In: *CVPR* (2022)
7. Corona, E., Pumarola, A., Alenya, G., Moreno-Noguer, F., Rogez, G.: Ganhand: Predicting human grasp affordances in multi-object scenes. In: *CVPR* (2020)
8. Coumans, E., Bai, Y.: Pybullet, a python module for physics simulation in robotics, games and machine learning. URL <http://pybullet.org> (2017)
9. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018)
10. Eppner, C., Mousavian, A., Fox, D.: Acronym: A large-scale grasp dataset based on simulation. In: *ICRA* (2021)
11. Fan, Z., Taheri, O., Tzionas, D., Kocabas, M., Kaufmann, M., Black, M.J., Hilliges, O.: Arctic: A dataset for dexterous bimanual hand-object manipulation. In: *CVPR* (2023)
12. Fang, H.S., Wang, C., Fang, H., Gou, M., Liu, J., Yan, H., Liu, W., Xie, Y., Lu, C.: Anygrasp: Robust and efficient grasp perception in spatial and temporal domains. *IEEE Transactions on Robotics* (2023)
13. Fang, H.S., Wang, C., Gou, M., Lu, C.: Graspnet-1billion: A large-scale benchmark for general object grasping. In: *CVPR* (2020)
14. Feix, T., Romero, J., Schmiedmayer, H.B., Dollar, A.M., Kragic, D.: The grasp taxonomy of human grasp types. *IEEE Transactions on human-machine systems* (2015)
15. Feng, Y., Lin, J., Dwivedi, S.K., Sun, Y., Patel, P., Black, M.J.: Chatpose: Chatting about 3d human pose. In: *CVPR* (2024)

16. Gao, D., Xiu, Y., Li, K., Yang, L., Wang, F., Zhang, P., Zhang, B., Lu, C., Tan, P.: Dart: Articulated hand model with diverse accessories and rich textures. *NeurIPS* (2022)
17. Garcia-Hernando, G., Yuan, S., Baek, S., Kim, T.K.: First-person hand action benchmark with rgb-d videos and 3d hand pose annotations. In: *CVPR* (2018)
18. Gemini: Introduction to Gemini. <https://deepmind.google/technologies/gemini/#introduction> (2023)
19. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. *NeurIPS* (2014)
20. Grady, P., Tang, C., Twigg, C.D., Vo, M., Brahmbhatt, S., Kemp, C.C.: Contactopt: Optimizing contact to improve grasps. In: *CVPR* (2021)
21. Hampali, S., Rad, M., Oberweger, M., Lepetit, V.: Honnotate: A method for 3d annotation of hand and object poses. In: *CVPR* (2020)
22. Hampali, S., Sarkar, S.D., Rad, M., Lepetit, V.: Keypoint transformer: Solving joint identification in challenging hands and object interactions for accurate 3d pose estimation. In: *CVPR* (2022)
23. Hasson, Y., Varol, G., Tzionas, D., Kalevatykh, I., Black, M.J., Laptev, I., Schmid, C.: Learning joint reconstruction of hands and manipulated objects. In: *CVPR* (2019)
24. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685* (2021)
25. Huang, J., Yong, S., Ma, X., Linghu, X., Li, P., Wang, Y., Li, Q., Zhu, S.C., Jia, B., Huang, S.: An embodied generalist agent in 3d world. In: *ICML* (2024)
26. Huang, S., Dong, L., Wang, W., Hao, Y., Singhal, S., Ma, S., Lv, T., Cui, L., Mohammed, O.K., Liu, Q., et al.: Language is not all you need: Aligning perception with language models. *arXiv preprint arXiv:2302.14045* (2023)
27. Hwangbo, J., Lee, J., Hutter, M.: Per-contact iteration method for solving contact dynamics. *IEEE Robotics and Automation Letters* (2018)
28. Jian, J., Liu, X., Li, M., Hu, R., Liu, J.: Affordpose: A large-scale dataset of hand-object interactions with affordance-driven hand pose. In: *ICCV* (2023)
29. Jiang, B., Chen, X., Liu, W., Yu, J., Yu, G., Chen, T.: Motiongpt: Human motion as a foreign language. *NeurIPS* (2023)
30. Jiang, H., Liu, S., Wang, J., Wang, X.: Hand-object contact consistency reasoning for human grasps generation. In: *ICCV* (2021)
31. Jin, S., Xu, J., Lei, Y., Zhang, L.: Reasoning grasping via multimodal large language model. *arXiv preprint arXiv:2402.06798* (2024)
32. Karunratanakul, K., Spurr, A., Fan, Z., Hilliges, O., Tang, S.: A skeleton-driven neural occupancy representation for articulated hands. In: *3DV* (2021)
33. Karunratanakul, K., Yang, J., Zhang, Y., Black, M.J., Muandet, K., Tang, S.: Grasping field: Learning implicit representations for human grasps. In: *3DV* (2020)
34. Kim, J., Kim, J., Na, J., Joo, H.: Parahome: Parameterizing everyday home activities towards 3d generative modeling of human-object interactions. *arXiv preprint arXiv:2401.10232* (2024)
35. Kudo, T., Richardson, J.: Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226* (2018)
36. Kwon, T., Tekin, B., Stühmer, J., Bogo, F., Pollefeys, M.: H2o: Two hands manipulating objects for first person interaction recognition. In: *ICCV* (2021)
37. Lakshmipathy, A.S., Feng, N., Lee, Y.X., Mahler, M., Pollard, N.: Contact edit: Artist tools for intuitive modeling of hand-object interactions. *ACM TOG* (2023)



38. Li, H., Lin, X., Zhou, Y., Li, X., Huo, Y., Chen, J., Ye, Q.: Contact2grasp: 3d grasp synthesis via hand-object contact constraint. In: IJCAI (2023)
39. Li, K., Yang, L., Lin, Z., Xu, J., Zhan, X., Zhao, Y., Zhu, P., Kang, W., Wu, K., Lu, C.: Favor: Full-body ar-driven virtual object rearrangement guided by instruction text. AAAI (2024)
40. Li, K., Yang, L., Zhen, H., Lin, Z., Zhan, X., Zhong, L., Xu, J., Wu, K., Lu, C.: Chord: Category-level hand-held object reconstruction via shape deformation. In: ICCV (2023)
41. Li, K., He, Y., Wang, Y., Li, Y., Wang, W., Luo, P., Wang, Y., Wang, L., Qiao, Y.: Videochat: Chat-centric video understanding. arXiv preprint arXiv:2305.06355 (2023)
42. Li, Y., Zhang, L., Qiu, Z., Jiang, Y., Li, N., Ma, Y., Zhang, Y., Xu, L., Yu, J.: Nimble: a non-rigid hand model with bones and muscles. ACM TOG (2022)
43. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. NeurIPS (2023)
44. Liu, M., Pan, Z., Xu, K., Ganguly, K., Manocha, D.: Deep differentiable grasp planner for high-dof grippers. arXiv preprint arXiv:2002.01530 (2020)
45. Liu, S., Zhou, Y., Yang, J., Gupta, S., Wang, S.: Contactgen: Generative contact modeling for grasp generation. In: ICCV (2023)
46. Liu, T., Liu, Z., Jiao, Z., Zhu, Y., Zhu, S.C.: Synthesizing diverse and physically stable grasps with arbitrary hand structures using differentiable force closure estimator. IEEE Robotics and Automation Letters (2021)
47. Liu, 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)
48. Liu, Y., Yang, H., Si, X., Liu, L., Li, Z., Zhang, Y., Liu, Y., Yi, L.: Taco: Benchmarking generalizable bimanual tool-action-object understanding. arXiv preprint arXiv:2401.08399 (2024)
49. Liu, Y., Liu, Y., Jiang, C., Lyu, K., Wan, W., Shen, H., Liang, B., Fu, Z., Wang, H., Yi, L.: Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In: CVPR (2022)
50. Lu, J., Kang, H., Li, H., Liu, B., Yang, Y., Huang, Q., Hua, G.: Ugg: Unified generative grasping. arXiv preprint arXiv:2311.16917 (2023)
51. Makoviychuk, V., Wawrzyniak, L., Guo, Y., Lu, M., Storey, K., Macklin, M., Hoeller, D., Rudin, N., Allshire, A., Handa, A., et al.: Isaac gym: High performance gpu-based physics simulation for robot learning. arXiv preprint arXiv:2108.10470 (2021)
52. Meta: Llama 3 (2024), <https://ai.meta.com/blog/meta-llama-3>
53. Miller, A.T., Allen, P.K.: Graspit! a versatile simulator for robotic grasping. IEEE Robotics & Automation Magazine (2004)
54. Mousavian, A., Eppner, C., Fox, D.: 6-dof graspnet: Variational grasp generation for object manipulation. In: ICCV (2019)
55. Newbury, R., Gu, M., Chumbley, L., Mousavian, A., Eppner, C., Leitner, J., Bohg, J., Morales, A., Asfour, T., Kragic, D., et al.: Deep learning approaches to grasp synthesis: A review. IEEE Transactions on Robotics (2023)
56. Nichol, A., Jun, H., Dhariwal, P., Mishkin, P., Chen, M.: Point-e: A system for generating 3d point clouds from complex prompts. arXiv preprint arXiv:2212.08751 (2022)
57. OpenAI: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
58. OpenAI: Gpt-4v(ision) system card (2023), <https://openai.com/research/gpt-4v-system-card>

59. Qin, Y., Wu, Y.H., Liu, S., Jiang, H., Yang, R., Fu, Y., Wang, X.: Dexmv: Imitation learning for dexterous manipulation from human videos. In: ECCV (2022)
60. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML (2021)
61. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research* (2020)
62. Razavi, A., Van den Oord, A., Vinyals, O.: Generating diverse high-fidelity images with vq-vae-2. *NeurIPS* (2019)
63. Romero, J., Tzionas, D., Black, M.J.: Embodied hands: Modeling and capturing hands and bodies together. *ACM TOG* (2017)
64. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017)
65. Sener, F., Chatterjee, D., Shelepov, D., He, K., Singhania, D., Wang, R., Yao, A.: Assembly101: A large-scale multi-view video dataset for understanding procedural activities. In: CVPR (2022)
66. Shadowrobot: Dexterous Hand Series. <https://www.shadowrobot.com/dexterous-hand-series/> (2005)
67. Sohn, K., Lee, H., Yan, X.: Learning structured output representation using deep conditional generative models. *NeurIPS* (2015)
68. Taheri, O., Choutas, V., Black, M.J., Tzionas, D.: Goal: Generating 4d whole-body motion for hand-object grasping. In: CVPR (2022)
69. Taheri, O., Ghorbani, N., Black, M.J., Tzionas, D.: Grab: A dataset of whole-body human grasping of objects. In: ECCV (2020)
70. Tang, C., Huang, D., Ge, W., Liu, W., Zhang, H.: Graspgpt: Leveraging semantic knowledge from a large language model for task-oriented grasping. *IEEE Robotics and Automation Letters* (2023)
71. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Baid, A., Brajberman, N., Bressand, P., Broun, S., et al.: Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288* (2023)
72. Turpin, D., Wang, L., Heiden, E., Chen, Y.C., Macklin, M., Tsogkas, S., Dickinson, S., Garg, A.: Grasp'd: Differentiable contact-rich grasp synthesis for multi-fingered hands. In: ECCV (2022)
73. Van Den Oord, A., Vinyals, O., et al.: Neural discrete representation learning. *NeurIPS* (2017)
74. Wan, W., Geng, H., Liu, Y., Shan, Z., Yang, Y., Yi, L., Wang, H.: Unidexgrasp++: Improving dexterous grasping policy learning via geometry-aware curriculum and iterative generalist-specialist learning. In: ICCV (2023)
75. Wu, Y., Wang, J., Zhang, Y., Zhang, S., Hilliges, O., Yu, F., Tang, S.: Saga: Stochastic whole-body grasping with contact. In: ECCV (2022)
76. Xie, W., Yu, Z., Zhao, Z., Zuo, B., Wang, Y.: Hmdo: Markerless multi-view hand manipulation capture with deformable objects. *Graphical Models* (2023)
77. Xu, R., Wang, X., Wang, T., Chen, Y., Pang, J., Lin, D.: Pointllm: Empowering large language models to understand point clouds. *arXiv preprint arXiv:2308.16911* (2023)
78. Xu, Y., Wan, W., Zhang, J., Liu, H., Shan, Z., Shen, H., Wang, R., Geng, H., Weng, Y., Chen, J., et al.: Unidexgrasp: Universal robotic dexterous grasping via learning diverse proposal generation and goal-conditioned policy. In: CVPR (2023)

79. Xue, L., Yu, N., Zhang, S., Li, J., Martín-Martín, R., Wu, J., Xiong, C., Xu, R., Niebles, J.C., Savarese, S.: Ulip-2: Towards scalable multimodal pre-training for 3d understanding. arXiv preprint arXiv:2305.08275 (2023)
80. Yang, L., Li, K., Zhan, X., Lv, J., Xu, W., Li, J., Lu, C.: Artiboost: Boosting articulated 3d hand-object pose estimation via online exploration and synthesis. In: CVPR (2022)
81. Yang, L., Li, K., Zhan, X., Wu, F., Xu, A., Liu, L., Lu, C.: Oakink: A large-scale knowledge repository for understanding hand-object interaction. In: CVPR (2022)
82. Yang, L., Zhan, X., Li, K., Xu, W., Zhang, J., Li, J., Lu, C.: Learning a contact potential field for modeling the hand-object interaction. IEEE Transactions on Pattern Analysis and Machine Intelligence (2024)
83. Yin, F., Chen, X., Zhang, C., Jiang, B., Zhao, Z., Fan, J., Yu, G., Li, T., Chen, T.: Shapegpt: 3d shape generation with a unified multi-modal language model. arXiv preprint arXiv:2311.17618 (2023)
84. Yu, X., Tang, L., Rao, Y., Huang, T., Zhou, J., Lu, J.: Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In: CVPR (2022)
85. Zhan, X., Yang, L., Zhao, Y., Mao, K., Xu, H., Lin, Z., Li, K., Lu, C.: Oakink2: A dataset of bimanual hands-object manipulation in complex task completion. In: CVPR (2024)
86. Zhang, H., Li, X., Bing, L.: Video-llama: An instruction-tuned audio-visual language model for video understanding. arXiv preprint arXiv:2306.02858 (2023)
87. Zhang, H., Ye, Y., Shiratori, T., Komura, T.: Manipnet: neural manipulation synthesis with a hand-object spatial representation. ACM TOG (2021)
88. Zhang, J., Zhang, Y., Cun, X., Huang, S., Zhang, Y., Zhao, H., Lu, H., Shen, X.: T2m-gpt: Generating human motion from textual descriptions with discrete representations. In: CVPR (2023)
89. Zheng, J., Zheng, Q., Fang, L., Liu, Y., Yi, L.: Cams: Canonicalized manipulation spaces for category-level functional hand-object manipulation synthesis. In: CVPR (2023)
90. Zheng, L., Chiang, W.L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E.P., Zhang, H., Gonzalez, J.E., Stoica, I.: Judging llm-as-a-judge with mt-bench and chatbot arena. arXiv preprint arXiv:2306.05685 (2023)
91. Zhou, Y., Barnes, C., Lu, J., Yang, J., Li, H.: On the continuity of rotation representations in neural networks. In: CVPR (2019)
92. Zhu, D., Chen, J., Shen, X., Li, X., Elhoseiny, M.: Minigpt-4: Enhancing vision-language understanding with advanced large language models. arXiv preprint arXiv:2304.10592 (2023)
93. Zhu, T., Wu, R., Lin, X., Sun, Y.: Toward human-like grasp: Dexterous grasping via semantic representation of object-hand. In: ICCV (2021)
94. Zhu, Z., Wang, J., Qin, Y., Sun, D., Jampani, V., Wang, X.: Contactart: Learning 3d interaction priors for category-level articulated object and hand poses estimation. arXiv preprint arXiv:2305.01618 (2023)