Multi-GraspLLM: A Multimodal LLM for Multi-Hand Semantic Guided Grasp Generation

Haosheng Li[1*], Weixin Mao[2*,†], Weipeng Deng[3], Chenyu Meng[1], Haoqiang Fan[4],
Tiancai Wang[4], Ping Tan[5], Hongan Wang[1], Xiaoming Deng[1‡]

[1]Institute of Software, Chinese Academy of Sciences
[2]Waseda University [3]University of Hong Kong [4]MEGVII Technology
[5]Hong Kong University of Science and Technology

Abstract

Multi-hand semantic grasp generation aims to generate feasible and semantically appropriate grasp poses for different robotic hands based on natural language instructions. Although the task is highly valuable, due to the lack of multi-hand grasp datasets with fine-grained contact description between robotic hands and objects, it is still a long-standing difficult task. In this paper, we present Multi-GraspSet, the first large-scale multi-hand grasp dataset with automatically contact annotations. Based on Multi-GraspSet, we propose Multi-GraspLLM, a unified language-guided grasp generation framework. It leverages large language models (LLM) to handle variable-length sequences, generating grasp poses for diverse robotic hands in a single unified architecture. Multi-GraspLLM first aligns the encoded point cloud features and text features into a unified semantic space. It then generates grasp bin tokens which are subsequently converted into grasp pose for each robotic hand via hand-aware linear mapping. The experimental results demonstrate that our approach significantly outperforms existing methods on Multi-GraspSet. More information can be found on our project page `https://multi-graspllm.github.io`.

1. Introduction

Grasp generation [24, 34, 38, 42, 44, 46, 47, 54, 55] is essential for robotic systems to interact with and manipulate their surroundings. Effective grasp generation is crucial for enhancing the versatility and adaptability of robotic systems, especially in complex scenarios where a wide range of tasks require precise handling. However, existing grasp generation methods mainly focus on predicting physically-
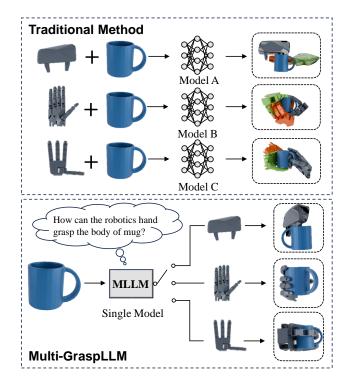

Figure 1. Multi-GraspLLM vs. traditional grasp generation method. Traditional grasp generation methods train separate models for each robotic hand, focusing primarily on generating physically stable grasps. In contrast, Multi-GraspLLM can use a single model to generate semantic-guided grasping poses adaptable to various robotic hands.

stable grasp poses using specialized models for individual robotic hands, while overlooking grasp generation across different robotic hands and the role of semantics in grasp generation.

In this work, we focus on a challenging problem of **multi-hand semantic-guided grasp generation**. This task aims to generate feasible and semantically appropriate grasp poses for multiple robotic hands through natural language

---

[*]Equal contribution.
[†]Project Lead.
[‡]Corresponding author.

1

instructions. Unlike traditional approaches that require specific models for each individual robotic hand (Figure 1), this task aims to generate grasp poses of multiple hands with a single model. This unified approach not only provides higher flexibility with only a single model to optimize but also enhances generalization through cross-hand learning. Moreover, while existing methods lack semantic understanding in grasp generation, this task incorporates semantic guidance to generate semantically appropriate grasps based on textual descriptions.

Generally, achieving this goal involves two major challenges. The first challenge is constructing a comprehensive dataset that covers diverse robotic hands and semantic descriptions, which is a non-trivial task. Among existing multi-hand datasets [16, 25, 38, 42] which primarily focus on physically stable grasps, only one dataset [44] provides rich contact semantic information. However, this dataset is limited to the Shadow Hand [31], and its retargeting approach based on human grasp data [50] cannot generalize well to other robotic hands with different link and joint structures, particularly for larger four-fingered systems like Allegro Hand [45]. Moreover, training a unified model for multiple robotic hands faces significant challenges when incorporating semantic guidance, as the same semantic instruction needs to be translated into different grasping behaviors for hands with distinctly different structures. Most semantic-guided methods [15, 43, 44] train a separate model for each hand. While DexGrasp-Diffusion [55] based on DexGraspNet [42] trains a unified diffusion model for different robotic hands, it lacks semantic guidance to grasp objects according to human common sense.

To address the first challenge, we propose Multi-GraspSet, the first large-scale multi-hand grasp dataset with contact annotations. Starting with pre-segmented objects [2, 50], we first generated a large number of physically stable grasps using the unified grasp generation approach [34, 42]. For each grasp case, we then added detailed contact annotations based on the segment annotation. Additionally, we leveraged LLM [1] to generate nearly 1M dialogue samples to support subsequent model training.

For the second challenge, based on the Multi-GraspSet, we design Multi-GraspLLM to generate precise, feasible grasp poses for multiple robotic hands. Multi-GraspLLM leverages the generalization capability and flexibility of large language models to handle variable-length inputs and ouputs, enabling a single model to produce poses for different robotic hands. The model takes a pair of point cloud and text instruction as input and outputs grasp angles for different robotic hands via a hand-aware linear mapping. For example, when given an instruction like "*How can we use the Allegro Hand to grasp a glass by its rim?*", Multi-GraspLLM can infer the optimal grasping angles for the

Allegro Hand [45] and adapt to other hands trained in the model, such as the Shadow Hand [31] and Panda Gripper [6].

Through extensive experimentation and evaluation of our dataset, we demonstrate that Multi-GraspLLM can successfully generate accurate grasp poses across multiple robotic hand types while maintaining precise control over individual finger positions. Quantitative results show significant improvements in both pose accuracy and grasp quality compared to existing methods. The model exhibits robust generalization capabilities across different grasp configurations while maintaining semantic consistency with natural language instructions.

Our key contributions can be summarized as follows.
1. We build Multi-GraspSet, the first large-scale multi-hand grasp dataset with contact annotations. Our dataset fills a gap in robotic hand grasp pose generation, particularly in multi-hand and semantic-guided grasping.
2. We propose Multi-GraspLLM, an LLM-based multi-hand grasp generation method that integrates LLM with cross-hand robotic grasp generation while ensuring semantically proper grasps.

2. Related Works

2.1. Robotics Grasp Datasets

Grasping datasets can generally be categorized into two types: physically stable and contact-aware. Most existing datasets focus on the former type. DexGraspNet [42, 54] and GenDexGrasp [16] use force-closure [20] as a metric for simulator optimization. Acronym [5] and MultiGripperGrasp [25] using the physical simulator [22] to filter out unstable grasps also support multiple robotic hands but have fewer objects. In contrast, contact-aware datasets transfer human hand poses to robotic hands. DexGYSNet [44] and Human-like-grasp [43] retarget the hand pose from the OakInk [50] dataset to the dexterous robotic hand and its own collected data with segmentation annotations. However, due to the difference in joints and links, DexGYSNet and Human-like-grasp typically only support those robotic hands resembling human hands.

2.2. Language-guided Grasp Generation

Although extensive research has been conducted on physically-stable grasp generation methods [11, 20, 24, 34, 38, 39, 42, 47, 54], there remains a significant gap in studies focused on language-guided functional grasp generation. The aim is to generate an appropriate grasp pose on a specific functional part of an object based on the description of the grasping scenario. Lerf-togo [30] and graspsplats [9] applied NERF [23] and 3DGS [13] to model 3D semantics scenes, allowing the gripper to grasp the specific object part based on language queries. Human-like-
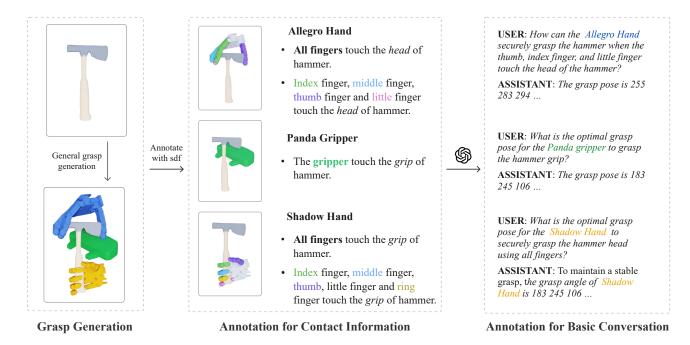
Figure 2. Multi-GraspSet Construction Process. The initial unified grasp generation produces physically stable grasps. Then, through two levels of annotations, we generate pair data containing basic conversation with corresponding grasp pose for each robotics hand.

grasp [43] trained a CVAE [33] model that allowed the generation of different functional grasping. However, it cannot support an open vocabulary. GaYs [44] addresses this limitation by training a language-conditioned diffusion model to generate dexterous grasp poses based on language descriptions. At the same time, large models greatly contribute to language-conditioned grasp generation. RealDex [21] leverages Gemini [36] to score generated grasps, filtering out those that satisfy the language description. SemGrasp [15] further extended this idea. It fine-tuned a large language model and used VQ-VAE [40] to discretize low-dimensional hand pose parameters. However, all of these methods only support single-type hand.

2.3. Multimodal LLMs in Robotics

Multimodal LLMs play a crucial role in reasoning for embodied intelligence, especially with the full development of models such as LLaVA [19] and LLaMA [37]. Several models [12, 29, 46, 48] leverage the reasoning capabilities of vision-language models [19] to identify grasping regions in scene images. Vision-language-action model [10, 14, 17, 18, 41, 57], built on the LLM architecture, enables the generation of discrete actions based on both visual and language input from a scene. Additionally, 3D large language models [32, 35, 49, 51] extend the structure of LLaVA by exchanging the 2D visual encoders with various 3D point cloud encoders such as PointNet [27, 28] or PointBert [52], which enable effective extraction of geometric features from unordered point sets and scene understand-

ing. Methods like [3, 7, 53, 56] refine tasks in embodied environments, while SemGrasp [15] focuses on more specific tasks, displaying human hand grasp poses based on inputs from object point cloud.

3. Multi-GraspSet Dataset

To our best knowledge, we proposed the first large-scale multi-hand grasp dataset with semantic contact annotation, namely Multi-GraspSet. This dataset was created through unified grasp generation and LLM assistance system. We detail the basic statistics in Sec. 3.1 then introduce the dataset construction process in Sec. 3.2.

3.1. Dataset statistics

Dataset	Hand	Object	Grasp	Con.
DexGYSNet [44]	1	1800	50k	-
CapGrasp [15]	1	1800	50k	280k
Multi-GraspSet	**3**	**2100**	**120k**	**1M**

Table 1. Comparison of the contact-aware grasp dataset. Grasp and Con. stand for the number of grasp pose and conversation.

Multi-GraspSet is a large-scale dataset that contains the grasp of three popular robotic hands, 2.1k object point clouds, 120k grasp pose pairs, and more than 1M conversations (Table 1). The dataset features two main types of annotations: **contact information** and **basic conversation**.

For contact information, we provide precise object contact annotations for each finger, and when all contact-involved fingers interact with the same object part, we provide a more general annotation emphasizing the number of fingers participating in the grasp, as illustrated in Figure 3. Moreover, to enable Multi-GraspLLM to generate grasps following natural language instructions, we construct basic conversations for each grasp pose with different control levels, comprising 5-10 diverse question-answer hand-aware basic conversations focusing on the grasp pose (Figure 6).

Allegro Hand

Index, ring, ring and thumb finger touch handle of bag

Four fingers touch handle of bag

Index, middle, ring, ring and thumb finger touch battery of power drill

All fingers touch the battery of power drill

Shadow Hand

Middle, ring finger touch the blade and thumb and index finger touch the handle of scissors

Index, middle, ring and thumb finger touch the lens of camera

All fingers touch the lens of camera

Panda Gripper

Gripper grasp the handle of knife

Gripper grasp the cap of bottle

Figure 3. Illustration of Multi-GraspSet. Our dataset includes grasp poses of common objects across three robotic hands with two types of contact annotations.

3.2. Dataset Construction

We first apply the unified grasp generation methods [34, 42] to generate physically stable grasp poses of collected meshes for dexterous hands and grippers. Then we design a contact information annotation method using signed distance fields (SDFs), which can produce fine-grained contact information between the finger and the object, accurately capturing the interaction between the gripper and the object. Finally, with the assistance of large language models [1], we generate a significant amount of conversational data, facilitating the dialogue-level understanding of robotic grasping.

3.2.1. Unified Grasp Generation

We generate grasp poses of three robotic hand through the unified grasp generation methods (Figure 2). We first collect 2,100 meshes from OakInk [50] and ShapeNet [2], applying convex decomposition to ensure that the meshes are watertight. Since we aim at multi-hand grasp generation, we then use DexGraspNet [42] to generate grasp poses

for dexterous robotic hands (Shadow Hand [31] and Allegro Hand [45]), and Contact-GraspNet [34] for grippers (Panda Gripper [6]). DexGraspNet [42] proposes a force-closure method for dexterous hands, working without learning. Contact-GraspNet [34] introduces a new end-to-end grasp generation network for grippers with strong generalization.

3.2.2. Annotation for Contact Information

Once the multi-hand grasp generation is complete, we use the Signed Distance Function (SDF) to annotate the contact information shown in Figure 2. Specifically, we assign a category label to each mesh face based on the segmentation annotations from the original dataset. For each object, we perform uniform sampling to generate a set of points on the object's surface. Next, we use Kaolin [26] to compute the SDF values from these points to the different links of the robotic hand. When the SDF value falls below a predefined threshold ϵ, we consider the corresponding link to be in contact with the object:

$$C(link_i) = \mathbb{I}(SDF(P_{link_i}|O) < \epsilon) \qquad (1)$$

where $C(link_i)$ indicates whether the i-th link is in contact with the object, O represents the object's mesh, and P denotes the point sampled from the robotic hand. Additionally, based on the category labels of object faces, we annotate which specific part of the object the link is in contact with. As briefly mentioned in Sec. 3.1, we use two types of annotation to describe the finger contacts: a detailed annotation that specifies individual fingers - "*Index, middle, thumb, and ring fingers contact the hammer's grip.*", and a general type for cases if all fingers contact the same part - "*Four fingers grasp the grip of the hammer.*". The general annotation is used to improve text quality and readability when fingers share the same contact location on the object.

3.2.3. Annotation for Basic Conversation

Inspired by SemGrasp [15], we develop an LLM-assisted Language Guidance Annotation system [1] to construct basic conversational datasets for robotic grasp tasks. We begin by prompting the LLM to generate templates that incorporate object names, robotic hand types, grasp parts, and contact details.

We categorize basic conversation into three types: low-level, middle-level, and high-level instructions, with levels of detail ranging from coarse to fine. Low-level instructions do not provide contact information and include only basic questions, such as "*How do you grasp the {object} using the {hand type}?*" Middle-level instructions specify the part to be grasped, for example, "*How do you grasp the {part} of the {object} using the {hand type}?*". High-level instructions include all relevant information, such as "*Demonstrate the ideal pose of the {hand type} to grasp the*

{object}: {contact info}." However, due to the complexity of our contact information compared to CapGrasp [15], fixed templates often lead to awkward phrasing. To address this, we use GPT-4o [1] to refine each sentence, ensuring varied and natural language throughout. Additional details about our annotation system are provided in the Supplementary Material.

4. Method

We propose Multi-GraspLLM, a framework based on a large language model [4] that generates appropriate grasp poses for different robotic hands by integrating point cloud data with natural language descriptions. We describe the detail of our framework in Sec. 4.1, then we show the data format and training strategy in Sec. 4.2 and Sec. 4.3, respectively.

4.1. Multi-GraspLLM

We initially apply a hand-aware discretization approach for grasp angles [14, 57]. These discretized grasp bins are then used to train the Multi-GraspLLM, enabling it to generate appropriate grasp poses across multiple robotic hands as shown in Figure 4.

Multi-hand Discretization. To transform the prediction of continuous grasp angle into a more tractable classification task for language models, we convert continuous angles into discrete tokens. Specifically, we uniformly discretize the grasp angles by dividing the valid range between the hand-specific angle lower bound L_{hand} and the upper bound U_{hand} into N bins, where both hand-specific bounds are computed across the entire dataset. The bin width W_{hand} is defined as $W_{hand} = (U_{hand} - L_{hand})/N$. Then we map a continuous angle p to its corresponding discrete bin B:

$$B = \frac{p - L_{hand}}{W_{hand}}, hand \in \{Allegro, Shadow, Panda\} \tag{2}$$

This discretization scheme is independently determined for each robotic hand based on its grasping dataset, ensuring efficient prediction while maintaining precision.

Grasp Pose Generation. Multi-GraspLLM comprises three components: (1) a 3D feature encoder f_{pc} that extract objects geometry information; (2) a modality adaptor f_m that aligns object geometry with language features; (3) a large language model m_{llm} for interpreting natural language grasp instructions and generating the grasp pose based on the 3D geometry of object.

Given an object point cloud $P \in \mathbb{R}^{8192}$ and a language description T, the point cloud encoder first encodes P into a sequence of feature tokens $s = f_{pc}(P)$, where $s \in \mathbb{R}^{512 \times 384}$. The modality adaptor then aligns these point cloud features into the language space $T_p = f_m(s)$, where $T_p \in \mathbb{R}^{512 \times 4096}$, to match the language features T_l encoded by the LLM language encoder. These two token sequences are concatenated to form a mixed token sequence $T_m = [T_p, T_l]$, which serves as input to the LLM backbone. The LLM backbone processes this sequence auto-regressively, generating outputs $O_i = m_{llm}(O_1, ..., O_{i-1})$ for the sequence $[O_1, O_2, ..., O_n]$. Finally, the de-tokenizer converts the output sequence into a grasp bin $B = [B_1, B_2, ...]$. Once the actions are processed into a sequence of bind, Multi-GraspLLM is trained with a standard next-token prediction objective, evaluating the Cross Entropy loss on the predicted grasp bin tokens and generated natural language to enhance the model's unified generalization capability.

We then de-discretize the grasp bin to grasp angle of specific robotic hand. Given the predicted grasp bin $B = [B_1, B_2, ...]$ from the de-tokenizer, the discrete bin index is converted linearly back to a continuous value as follows:

$$[T, R, \theta] = L_{hand} + B_i W_{hand} \tag{3}$$

where $[T; R] \in \mathbb{R}^6$ represents the 6-D pose of the robotic hand's wrist, and $\theta \in \mathbb{R}^d$ denotes the joint angles specific to different robotic hands ($d = 6$ for Panda Gripper [6], $d = 22$ for Allegro Hand [45], and $d = 28$ for Shadow Hand [31]). The detailed implementation is provided in the supplementary material.

4.2. Data Format

As shown in Figure 5, we introduce three special tokens in the Vicuna [4] backbone fine-tuning: (1) a Robotics hand special token for hand identification, (2) a Scale special token for point cloud size normalization, and (3) a Grasp special bin token mapping to 512 discretization bins. These tokens work together to ensure accurate information processing during both tokenization and de-tokenization.

With the help of these specialized tokens, we combined and refined the basic conversations from Multi-GraspSet and the caption task. The final dataset consists of four types of conversations, as illustrated in Figure 5: (1) **Single-Round Grasp Generation**; (2) **Multi-Round Mix**; (3) **Multi-Round Grasp Generation**. The first type involves single-round interactions, where the model predicts grasping angles based on simple queries like "*How can the {robotic hand} grasp the object?*". The latter two types feature multi-round interactions. "Multi-Round Mix" starts with object class description and follows with grasp prediction. The "Multi-Round Grasp Generation" supports more diverse dialogues involving different robotic hands and requiring sophisticated grasp predictions, going beyond the basic approach. We purposely incorporated captioning tasks to enhance the model's 3D geometric understanding and reduce its reliance on text-only inputs. Our ablation studies in Sec. 5.4 validate that this strategy significantly improves performance, achieving a balanced interpretation of both spatial geometry and grasping actions.
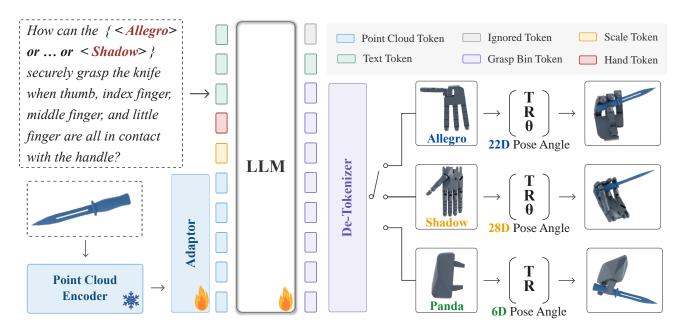
Figure 4. Multi-GraspLLM model. The point encoder extracts point clouds from objects and maps them with language descriptions into the same latent space. The LLM backbone then generate grasp bin tokens as output. Finally, we convert these grasp bin tokens into corresponding grasp angles for each robotic hand.
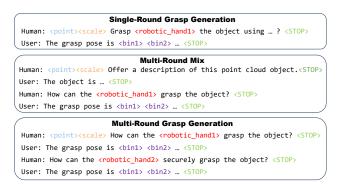

Figure 5. The template of training data used to train the Multi-GraspLLM.

4.3. Training Strategy

To effectively train our model, we employ a two-stage training procedure consisting of multimodal alignment and instruction tuning.

Stage1: Multimodal Alignment. In multimodal alignment, we aim to have the adaptor map point cloud modality data and language modality data into a space that is as close as possible. We freeze the LLM backbone and point cloud encoder while fine-tuning only the modality adaptor. The fine-tuning process incorporates single-round conversations "Single-Round Grasp Generation" described in Sec. 4.2.

Stage2: Instruction Tuning. In instruction tuning, we want our LLM to thoroughly understand the grasp generation task. We always keep the point cloud encoder weights

frozen, and continue to update both the pre-trained weights of the adaptor and LLM in Multi-GraspLLM. In this stage, we tend to use complex conversations that consist of all types as described in Sec. 4.2.

5. Experiments

5.1. Dataset and Metric

Our Multi-GraspSet dataset is split into three parts: 80% for training, 10% for validation, and 10% for testing. The point clouds in the test set are completely separate from those used in training and validation.

Multi-GraspLLM is evaluated in two key aspects: grasp intention and physical stability. To evaluate grasp intention accuracy, we employ **Chamfer distance (CD)** (cm), which quantifies the spatial alignment between the predicted and ground truth robotics hand point clouds. This metric measures how well our inferred grasps match the intended grasp locations on objects as specified in the descriptions, where lower values indicate better alignment. To evaluate grasp quality, we employ three metrics: the **maximum penetration distance (Pen-d)** (cm) between the gripper and target object measures physical feasibility, the **grasp success rate (Suc.)** in Isaac Sim and the **Q1 metric** which is intuitively the norm of the smallest wrench that can destabilize the grasp [42]. More details are in our supplementary material.

5.2. Implementation Details

For constructing Multi-GraspSet, we employ DexGrasp-Net [42] and Contact-GraspNet [34] as our grasp generation

6

method	Allegro				Shadow				Panda				Avg.			
	CD↓	Pen-d↓	Suc. ↑	Q1↑	CD ↓	Pen-d↓	Suc. ↑	Q1↑	CD↓	Pen-d↓	Suc.↑	Q1↑	CD↓	Pen-d ↓	Suc. ↑	Q1↑
GraspTTA [11]	1.31	1.12	0.26	0.075	1.20	1.18	0.31	0.062	/	/	/	/	1.27	1.15	0.29	0.068
SceneDiffuser [8]	0.74	1.04	0.31	0.086	1.02	0.86	0.37	0.081	/	/	/	/	0.88	0.96	0.35	0.084
6Dof-GraspNet [24]	/	/	/	/	/	/	/	/	0.59	0.41	0.40	/	/	/	/	/
Contact-GraspNet [34]	/	/	/	/	/	/	/	/	0.62	0.43	0.46	/	/	/	/	/
Multi-GraspLLM-mix	0.43	**0.97**	0.29	0.092	0.48	0.80	**0.42**	**0.089**	0.29	**0.37**	0.45	/	0.40	0.71	0.38	**0.091**
Multi-GraspLLM-split	**0.39**	0.99	**0.32**	**0.094**	**0.45**	**0.72**	0.40	0.083	**0.26**	0.42	**0.48**	/	**0.37**	**0.70**	**0.40**	0.089

Table 2. Comparison of SOTA and Multi-GraspLLM. '-mix' denotes models trained on mixed datasets, while '-split' indicates models trained separately for each robotic hand.

methods. For DexGraspNet, we generate 128 grasp poses for the Allegro Hand [45] and 160 for the Shadow Hand [31] per object, using Adam optimizer with 6000 epochs. For Contact-GraspNet [34], we quickly sample 256 points from each point cloud and generate 200 grasps.

During training, we perform multimodal alignment for 3 epochs with a learning rate of 2e-3, followed by instruction tuning for another 3 epochs with a learning rate of 2e-5. For each object, we uniformly sample 8,192 points from its mesh to generate the point cloud, which is then tokenized into 512 tokens. The grasp angles are discretized into 384 bins, with their upper and lower bounds for different robotic hands determined by their entire dataset.

5.3. Comparison with State-of-the-Art Methods

We compared our Multi-GraspLLM method with existing open-source SOTA approaches. Since there is currently no open-source method for multi-dexterous hand and gripper grasping generation, we conducted separate comparisons on corresponding end-effectors: SceneDiffuser [8] and GraspTTA [11] were evaluated on dexterous hands, both trained on our dataset, while Contact-GraspNet [34] and 6dof-GraspNet [24] were assessed on parallel grippers. The main results are shown in Table 2, "GraspLLM-split" means our Multi-GraspLLM model trained on the dataset separately for each manipulator, while "-mix" refers to mixed training. We can see that mixed training performs slightly worse than separate training, but the drop in performance is minor. Whether trained separately or mixed, both methods outperform all existing baselines, especially in intention accuracy, with better physical stability and less penetration. Unlike DexGraspNet [42], which adds specific loss functions for physical stability, and the two-stage method "Grasp as You Say" [44], our approach is end-to-end without any extra loss functions. Although the physical stability of our method does not exceed the limits of the training dataset, it still outperforms all other compared methods.

As shown in Figure 6, Multi-GraspLLM can generate different robotic hand grasping poses based on instructions with varying levels of contact information. Our approach supports grasp generation at different instruction levels. For the low-level instruction, model generates grasps without any contact information, while mid-level focuses on the object part information. The high-level instruction utilizes finger contact information, enabling our model to simply control individual fingers.

5.4. Ablation Study

To verify the effectiveness of each component of our method, we conducted ablation studies, examining the impact of each special token and the presence of two-stage training. Additionally, we explored the effects of grasp bin length and explored how the structure and size of training data affect the model's performance.

	CD↓	Pen-d↓	Suc.↑	Q1↑
Multi-GraspLLM	**0.40**	**0.71**	**0.38**	**0.091**
w/o hand token	0.43	0.78	0.38	0.088
w/o scale	0.49	1.01	0.32	0.081
w/o grasp	0.45	1.09	0.32	0.075
w/o 2-stage	0.47	1.11	0.34	0.071

Table 3. Ablation study of our Multi-GraspLLM grasp generation.

We first performed ablation on each special token. As shown in Table 3, the two-stage training and the existence of grasp bin token significantly impact performance. A single-stage model struggles to respond effectively to newly introduced grasp bin tokens it has not encountered before. Due to the incomplete encoding vocabulary for numbers in Vicuna's [4] tokenizer, larger numbers without grasp bin token such as "400" may be decoded as separate tokens ["40","0"], which further hinders the model's grasp bin comprehension. Additionally, since PointBERT [52] normalizes point clouds, providing a scale token helps the model better understand point cloud shapes. Lastly, the robotic hand special token has minimal impact in our case, as we only use three types of robotic hands, which the model can memorize by name. However, with a larger variety of hands, the robotic hand special token would play a more significant role.
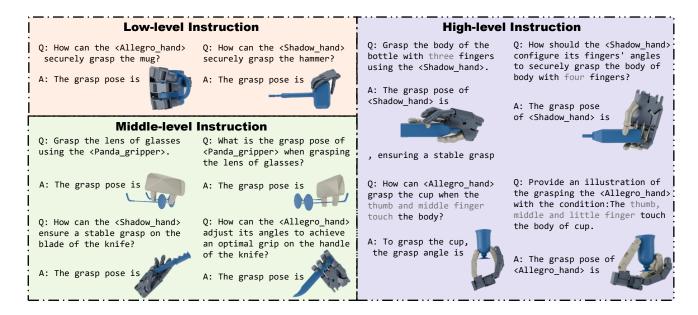
Figure 6. Visualization of the grasp pose generated by Multi-GraspLLM.

Number	CD↓	Pen-d↓	Suc. ↑	Q1↑
512bin	0.41	0.79	**0.39**	0.088
384bin	**0.40**	**0.71**	0.38	**0.091**
256bin	0.45	0.8	0.31	0.086

Table 4. Effect of grasp bin number.

Next, we examined the effect of the number of grasp bins. The choice of bin number presents a trade-off: smaller bin numbers (resulting in larger bin widths) provide higher resolution, but make prediction more challenging. Larger bin numbers simplify prediction but reduce the precision of multi-hand reconstruction. We therefore selected a moderate bin width to balance these competing factors.
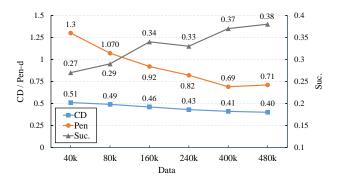

Figure 7. Ablation on training data for modality alignment.

We also analyzed the effect of the size of the training data in modality alignment and structure of data in instruction tuning, as shown in Figure 7 and Table 5. We gradually reduced the amount of modality alignment data to test our alignment method. As shown in Figure 7, using more data improves the performance of our model in intention understanding (lower CD) and grasp quality (higher Q1 and Suc.).

We then performed an ablation study on the composition of the instruction tuning data. As described in Section 4.2, our instruction tuning data consists of three types. "Single-Grasp" represents the "Single-Round Grasp Generation", and "Mix" is the abbreviation of "Multi-Round Mix". "Multi-Grasp" represents multi-round conversations for generating grasps, corresponding to "Multi-Round Grasp Generation" in Section 4.2. As shown in Table 5, more complex and multi-round conversations used during fine-tuning lead to better performance. Due to the introduction of caption data, the model can be prevented from excessively focusing all attention on natural language while ignoring the geometric features of point clouds, which leads to improved grasp prediction intention.

Single-Grasp	Mix	Multi-Grasp	CD↓	Pen-d↓	Suc.↑	Q1↑
✓			0.43	0.73	0.35	0.086
✓	✓		0.42	0.76	0.36	0.089
✓	✓	✓	**0.40**	**0.71**	**0.38**	**0.091**

Table 5. Ablation on training data for instruction tuning.

6. Conclusion

In this work, We present the first multi-hand grasp dataset with semantic guidance, along with a multimodel LLM for multi-hand grasp generation named Multi-GraspLLM. It can generate grasp poses for different robotic hands based on linguistic grasp descriptions. Our approach fills a gap

in the cross-hand semantic grasp generation field. In the future, we plan to collect larger data using more types of robotic hands and objects, which can further enhance the ability of Multi-GraspLLM to generalize to real-world scenarios, ultimately driving advancements in multi-robot collaboration and human-robot interaction.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 2, 4, 5

[2] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *CoRR*, abs/1512.03012, 2015. 2, 4

[3] Sijin Chen, Xin Chen, Chi Zhang, Mingsheng Li, Gang Yu, Hao Fei, Hongyuan Zhu, Jiayuan Fan, and Tao Chen. Ll3da: Visual interactive instruction tuning for omni-3d understanding reasoning and planning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26428–26438, 2024. 3

[4] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023. 5, 7

[5] Clemens Eppner, Arsalan Mousavian, and Dieter Fox. Acronym: A large-scale grasp dataset based on simulation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6222–6227. IEEE, 2021. 2

[6] Franka Emika GmbH. Franka emika robots. https://franka.de, 2016. 2, 4, 5

[7] Jiangyong Huang, Silong Yong, Xiaojian Ma, Xiongkun Linghu, Puhao Li, Yan Wang, Qing Li, Song-Chun Zhu, Baoxiong Jia, and Siyuan Huang. An embodied generalist agent in 3d world. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. 3

[8] 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*, pages 16750–16761, 2023. 7

[9] Mazeyu Ji, Ri-Zhao Qiu, Xueyan Zou, and Xiaolong Wang. Graspsplats: Efficient manipulation with 3d feature splatting. *arXiv preprint arXiv:2409.02084*, 2024. 2

[10] Fan Jia, Weixin Mao, Yingfei Liu, Yucheng Zhao, Yuqing Wen, Chi Zhang, Xiangyu Zhang, and Tiancai Wang. Adriver-i: A general world model for autonomous driving. *CoRR*, abs/2311.13549, 2023. 3

[11] 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*, pages 11107–11116, 2021. 2, 7

[12] Shiyu Jin, Jinxuan Xu, Yutian Lei, and Liangjun Zhang. Reasoning grasping via multimodal large language model. *arXiv preprint arXiv:2402.06798*, 2024. 3

[13] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2

[14] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024. 3, 5

[15] Kailin Li, Jingbo Wang, Lixin Yang, Cewu Lu, and Bo Dai. Semgrasp : Semantic grasp generation via language aligned discretization. In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part II*, pages 109–127. Springer, 2024. 2, 3, 4, 5

[16] Puhao Li, Tengyu Liu, Yuyang Li, Yiran Geng, Yixin Zhu, Yaodong Yang, and Siyuan Huang. Gendexgrasp: Generalizable dexterous grasping. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8068–8074. IEEE, 2023. 2

[17] Xinghang Li, Minghuan Liu, Hanbo Zhang, Cunjun Yu, Jie Xu, Hongtao Wu, Chilam Cheang, Ya Jing, Weinan Zhang, Huaping Liu, Hang Li, and Tao Kong. Vision-language foundation models as effective robot imitators. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. 3

[18] Xinghang Li, Minghuan Liu, Hanbo Zhang, Cunjun Yu, Jie Xu, Hongtao Wu, Chilam Cheang, Ya Jing, Weinan Zhang, Huaping Liu, Hang Li, and Tao Kong. Vision-language foundation models as effective robot imitators. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. 3

[19] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. 3

[20] Tengyu Liu, Zeyu Liu, Ziyuan Jiao, Yixin Zhu, and Song-Chun Zhu. Synthesizing diverse and physically stable grasps with arbitrary hand structures using differentiable force closure estimator. *IEEE Robotics and Automation Letters*, 7(1): 470–477, 2021. 2

[21] Yumeng Liu, Yaxun Yang, Youzhuo Wang, Xiaofei Wu, Jiamin Wang, Yichen Yao, Sören Schwertfeger, Sibei Yang, Wenping Wang, Jingyi Yu, Xuming He, and Yuexin Ma. Realdex: Towards human-like grasping for robotic dexterous hand. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024, Jeju, South Korea, August 3-9, 2024*, pages 6859–6867. ijcai.org, 2024. 3

[22] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, and Gavriel State. Isaac gym: High performance GPU based physics simulation for robot learning. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, 2021. 2

[23] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2

[24] Arsalan Mousavian, Clemens Eppner, and Dieter Fox. 6-dof graspnet: Variational grasp generation for object manipulation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2901–2910, 2019. 1, 2, 7

[25] Luis Felipe Casas Murrilo, Ninad Khargonkar, Balakrishnan Prabhakaran, and Yu Xiang. Multigrippergrasp: A dataset for robotic grasping from parallel jaw grippers to dexterous hands. *arXiv preprint arXiv:2403.09841*, 2024. 2

[26] NVIDIA. Kaolin: A pytorch library for accelerating 3d deep learning research. https://github.com/NVIDIAGameWorks/kaolin, 2019. 4

[27] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 3

[28] 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*, 30, 2017. 3

[29] Shengyi Qian, Weifeng Chen, Min Bai, Xiong Zhou, Zhuowen Tu, and Li Erran Li. Affordancellm: Grounding affordance from vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7587–7597, 2024. 3

[30] Adam Rashid, Satvik Sharma, Chung Min Kim, Justin Kerr, Lawrence Yunliang Chen, Angjoo Kanazawa, and Ken Goldberg. Language embedded radiance fields for zero-shot task-oriented grasping. In *7th Annual Conference on Robot Learning*, 2023. 2

[31] Shadow Robot Company. Dexterous hand series. https://www.shadowrobot.com/dexterous-hand-series/, 2005. 2, 4, 5, 7

[32] Yawar Siddiqui, Antonio Alliegro, Alexey Artemov, Tatiana Tommasi, Daniele Sirigatti, Vladislav Rosov, Angela Dai, and Matthias Nießner. Meshgpt: Generating triangle meshes with decoder-only transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19615–19625, 2024. 3

[33] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28, 2015. 3

[34] Martin Sundermeyer, Arsalan Mousavian, Rudolph Triebel, and Dieter Fox. Contact-graspnet: Efficient 6-dof grasp gen-

eration in cluttered scenes. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13438–13444. IEEE, 2021. 1, 2, 4, 6, 7

[35] Yuan Tang, Xu Han, Xianzhi Li, Qiao Yu, Yixue Hao, Long Hu, and Min Chen. Minigpt-3d: Efficiently aligning 3d point clouds with large language models using 2d priors. *arXiv preprint arXiv:2405.01413*, 2024. 3

[36] Gemini Team. Gemini: A family of highly capable multimodal models. 2024. 3

[37] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 3

[38] Dylan Turpin, Liquan Wang, Eric Heiden, Yun-Chun Chen, Miles Macklin, Stavros Tsogkas, Sven Dickinson, and Animesh Garg. Grasp'd: Differentiable contact-rich grasp synthesis for multi-fingered hands. In *European Conference on Computer Vision*, pages 201–221. Springer, 2022. 1, 2

[39] Dylan Turpin, Tao Zhong, Shutong Zhang, Guanglei Zhu, Eric Heiden, Miles Macklin, Stavros Tsogkas, Sven Dickinson, and Animesh Garg. Fast-grasp'd: Dexterous multifinger grasp generation through differentiable simulation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8082–8089. IEEE, 2023. 2

[40] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 3

[41] Quan Vuong, Sergey Levine, Homer Rich Walke, Karl Pertsch, Anikait Singh, Ria Doshi, Charles Xu, Jianlan Luo, Liam Tan, Dhruv Shah, et al. Open x-embodiment: Robotic learning datasets and rt-x models. In *Towards Generalist Robots: Learning Paradigms for Scalable Skill Acquisition@ CoRL2023*, 2023. 3

[42] Ruicheng Wang, Jialiang Zhang, Jiayi Chen, Yinzhen Xu, Puhao Li, Tengyu Liu, and He Wang. Dexgraspnet: A large-scale robotic dexterous grasp dataset for general objects based on simulation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11359–11366. IEEE, 2023. 1, 2, 4, 6, 7

[43] Wei Wei, Peng Wang, Sizhe Wang, Yongkang Luo, Wanyi Li, Daheng Li, Yayu Huang, and Haonan Duan. Learning human-like functional grasping for multi-finger hands from few demonstrations. *IEEE Transactions on Robotics*, 2024. 2, 3

[44] Yi-Lin Wei, Jian-Jian Jiang, Chengyi Xing, Xiantuo Tan, Xiao-Ming Wu, Hao Li, Mark Cutkosky, and Wei-Shi Zheng. Grasp as you say: Language-guided dexterous grasp generation. *arXiv preprint arXiv:2405.19291*, 2024. 1, 2, 3, 7

[45] Wonik Robotics. Allegro hand. https://www.allegrohand.com, 2016. 2, 4, 5, 7

[46] William Xie, Maria Valentini, Jensen Lavering, and Nikolaus Correll. Deligrasp: Inferring object properties with llms for adaptive grasp policies. In *8th Annual Conference on Robot Learning*, 2024. 1, 3

[47] Guo-Hao Xu, Yi-Lin Wei, Dian Zheng, Xiao-Ming Wu, and Wei-Shi Zheng. Dexterous grasp transformer. In *Proceed-*

ings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 17933–17942, 2024. 1, 2

[48] Jinxuan Xu, Shiyu Jin, Yutian Lei, Yuqian Zhang, and Liangjun Zhang. Reasoning tuning grasp: Adapting multi-modal large language models for robotic grasping. In *2nd Workshop on Language and Robot Learning: Language as Grounding*, 2023. 3

[49] Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiang-miao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part XXV*, pages 131–147. Springer, 2024. 3

[50] 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*, pages 20953–20962, 2022. 2, 4

[51] Fukun Yin, Xin Chen, Chi Zhang, Biao Jiang, Zibo Zhao, Ji-ayuan Fan, Gang Yu, Taihao Li, and Tao Chen. Shapegpt: 3d shape generation with a unified multi-modal language model. *arXiv preprint arXiv:2311.17618*, 2023. 3

[52] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19313–19322, 2022. 3, 7

[53] Yangbin Yu, Qin Zhang, Junyou Li, Qiang Fu, and De-heng Ye. Affordable generative agents. *arXiv preprint arXiv:2402.02053*, 2024. 3

[54] Jialiang Zhang, Haoran Liu, Danshi Li, XinQiang Yu, Hao-ran Geng, Yufei Ding, Jiayi Chen, and He Wang. Dexgrasp-net 2.0: Learning generative dexterous grasping in large-scale synthetic cluttered scenes. In *8th Annual Conference on Robot Learning*. 1, 2

[55] Zhengshen Zhang, Lei Zhou, Chenchen Liu, Zhiyang Liu, Chengran Yuan, Sheng Guo, Ruiteng Zhao, Marcelo H Ang Jr, and Francis EH Tay. Dexgrasp-diffusion: Diffusion-based unified functional grasp synthesis pipeline for multi-dexterous robotic hands. *arXiv preprint arXiv:2407.09899*, 2024. 1, 2

[56] Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla: A 3d vision-language-action generative world model. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. Open-Review.net, 2024. 3

[57] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023. 3, 5