DexVLG: Dexterous Vision-Language-Grasp Model at Scale

Jiawei He[1,2*], Danshi Li[2*], Xinqiang Yu[2*], Zekun Qi[2,3], Wenyao Zhang[2,6,7], Jiayi Chen[2,4], Zhaoxiang Zhang[5✉], Zhizheng Zhang[2✉], Li Yi[3✉], He Wang[1,2,4✉]
[1] Beijing Academy of Artificial Intelligence, [2] Galbot, [3] Tsinghua University,
[4] Peking University, [5] Institute of Automation, Chinese Academy of Sciences,
[6] Shanghai Jiao Tong University, [7] Eastern Institute of Technology
[*]: Equal Contribution, [✉] Corresponding authors

Project Page: https://jaweihe.com/dexvlg

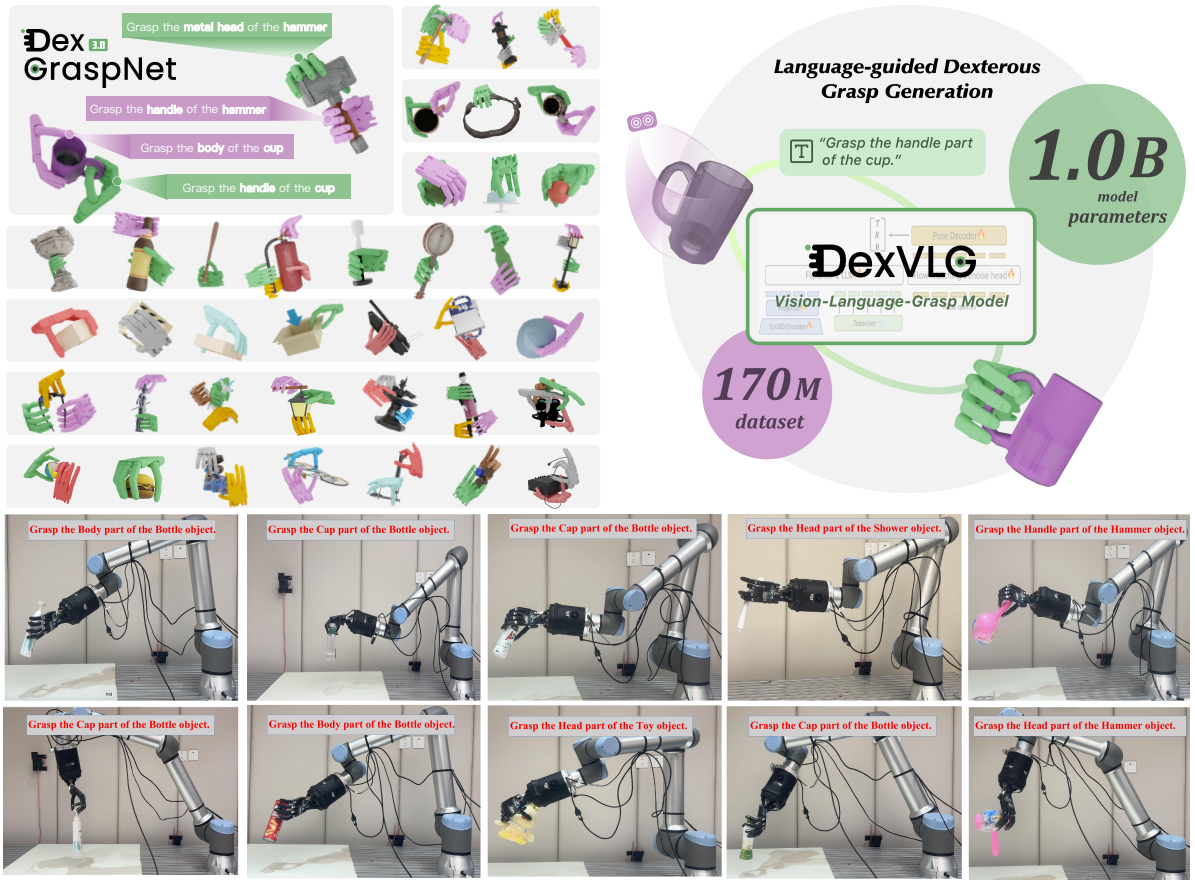

Figure 1. **Overview**. Our main contributions are twofold: First, we constructed a large-scale synthetic dexterous grasping dataset called DexGraspNet3.0, which contains grasp poses with captions describing the grasped part and style. Second, we trained a language-instructed grasp pose prediction model using the DexGraspNet3.0 dataset, called DexVLG. This model can generate language-aligned and generalizable grasping poses for different objects in real-world experiments.

Abstract

*As large models gain traction, vision-language models are enabling robots to tackle increasingly complex tasks. However, limited by the difficulty of data collection, progress has mainly focused on controlling simple gripper end-effectors. There is little research on functional grasping with large models for human-like dexterous hands. In this paper, we introduce **DexVLG**, a large **V**ision-**L**anguage-**G**rasp model for **Dexterous** grasp pose prediction aligned with language*

*instructions using single-view RGBD input. To accomplish this, we generate a dataset of 170 million dexterous grasp poses mapped to semantic parts across 174,000 objects in simulation, paired with detailed part-level captions. This large-scale dataset, named **DexGraspNet 3.0**, is used to train a VLM with a flow-matching-based pose head producing instruction-aligned grasp poses for tabletop objects. To evaluate DexVLG's performance, we create benchmarks in simulations and conduct real-world experiments. Extensive experiments demonstrate DexVLG's strong zero-shot generalization capabilities, achieving an over 76% zero-shot execution success rate and state-of-the-art part-grasp accuracy in simulation, as well as successful part-aligned grasps on physical objects in real-world scenarios.*

1. Introduction

To unleash the potential of intelligent abilities for a robot, recent advances in large vision-language-action (VLA) models [3, 18, 40] have shown a promising way. They have demonstrated strong generalizability on many complex robotic tasks across diverse scenarios in the real world. The key reason for their success is the large model capacity and training dataset: their model typically has billion-level parameters and is trained on billion-level robotic datasets.

However, those large VLA models are currently limited to parallel grippers and cannot control a dexterous hand. The main reason is the lack of data for dexterous grasping. Some works retarget from human motion [26, 48, 54] and teleoperate a real robot [35, 45] to collect data, but they all require significant human effort. Some works [6, 46, 57] use analytic-based methods to quickly synthesize a large-scale dexterous grasp dataset, but they are semantic-unaware and thus cannot perform functional grasps like humans do. For example, humans usually hold a hammer by the handle to use it, but grip the metal part when handing it to someone else. Recent research on functional dexterous grasp [14, 16, 49] can only use small-scale datasets, greatly limiting the model capacity and generalizability.

To address the data challenge, in this work, we propose a large-scale part-aware functional dexterous grasp dataset, named **DexGraspNet 3.0**. Our dataset contains 170M dexterous grasp poses on 174k objects from the Objaverse [9] dataset. Each grasp pose is validated in a physics-based simulation and paired with captions describing the grasped part name and grasp style. To build this, we follow the DexGraspNet series of works [46, 57] for efficient grasp synthesis, and introduce part-aware energies to make each grasp semantically distinguishable. We also leverage state-of-the-art object part understanding models like SAMesh [38] and GPT-4o for part segmentation and captioning.

Powered by DexGraspNet 3.0, we developed **DexVLG**, a large vision-language-grasp model. DexVLG takes a language instruction and a single-view colored point cloud of tabletop objects as input and generates dexterous grasp poses based on the instruction. DexVLG leverages multiple pre-trained foundation models to extract vision-language features and employs a flow-matching denoising paradigm to predict grasp poses. With billions of parameters, the model is fine-tuned end-to-end on our large-scale dataset.

To evaluate the performance of DexVLG, we perform experiments in both simulation and the real world. We first build a benchmark for part-aware dexterous grasping in Isaac Gym [29], with novel metrics that evaluate the part-alignment of dexterous grasp poses. Several baselines are compared to show the superiority of our model. DexVLG outperforms baselines on all benchmarks and achieves over 76% grasp success rate. We also demonstrate successful real-world executions predicted by DexVLG.

To summarize, our contributions are as follows:

- We introduce DexGraspNet3.0, a large-scale dataset containing 170M part-aligned dexterous grasp poses on 174k objects, each annotated with semantic captions.
- We propose a VLM named DexVLG to generate language-instructed dexterous grasp poses end-to-end.
- We curate benchmarks and conduct extensive experiments to assess DexVLG in simulation and real world.

2. Related Work

2.1. 3D Part Segmentation

3D part segmentation splits a 3D object into distinct components. Early methods [27, 60] rely on human-annotated small datasets and struggle to generalize beyond the training distribution. The PartSLIP series [24, 63] pioneers the application of 2D VLMs to 3D part segmentation. More recent works [28, 55] pretrain 3D VLMs on the huge Objaverse [9] dataset and demonstrate much stronger generalizability. SAMesh [38] is a zero-shot geometric part segmentation method on mesh, combining traditional shape geometric features and learning-based SAM. In this paper, we utilize the geometric SAMesh method to generate separated parts and VLMs [30, 39] to assign semantics to parts.

2.2. Dexterous Grasp Synthesis

Many recent works on dexterous grasping are semantic-unaware. Some analytic-based synthesis works [5–7, 19, 25, 41, 42] study the efficient generation of grasps by optimizing a differentiable grasp quality metric, given the object mesh. These methods enable the generation of million-level dexterous grasp datasets like DexGraspNet 1.0 [46] and 2.0 [57].Others [8, 17, 51, 52, 59] study to perform supervised learning on grasping datasets, using conditional generative models like CVAE, diffusion model and normalizing flows. Reinforcement learning [43, 56] is another hot topic for dexterous grasping, but up to now, most results are

Data Source	Dataset	Hand	Object	Grasp	Caption	Simulation Check	Semantics	Part
Real-World	OakInk [54]	MANO	100	50k	None		√	√
	DexGYSNet [48]	Shadow	1800	50k	50k		√	
	SemGrasp [21]	MANO	1800	50k	280k	√	√	
Simulation	Grasp'D [42]	Shadow,Allegro	2408	1M	None	√		
	DexGraspNet [46]	Shadow	5355	1.32M	None	√		
	DexGraspNet 2.0 [57]	Leap	88	45.04M	None	√		
	BoDex [6]	Shadow	2397	7.17M	None	√		
	Ours	**Shadow**	**174k**	**170M**	**170M**	√	√	√

Table 1. **Comparison of DexGraspNet 3.0 with existing dexterous grasping datasets.** All of our grasp poses are validated in IsaacGym [29] and paired with part annotation and language captions.

only presented in simulation. A few works study semantic-aware dexterous grasping [2, 14, 16, 21, 48, 49], but their studied object instances and categories are limited in scale. Our work, in contrast, introduces the semantic-aware energy to the advanced analytic-based method [5, 6, 57], synthesizing a large-scale functional dexterous grasp dataset.

2.3. Vision-Language models for Robotic Action

Using vision-language models (VLM) to control robot action is an emerging research area. One way to achieve this is to decompose robot manipulation into a series of VQA tasks and execute 6D waypoints planned by a VLM [23, 44]. This is infeasible for dexterous grasping because a 6D end effector pose cannot fully characterize high-dimensional dexterous hand states. Another way is to learn generalizable action priors from large-scale robot trajectory datasets, which produces vision-language-action(VLA) models [4, 18, 40, 58]. Existing dexterous grasp datasets are limited in scale and cannot support learning a VLA. The most relevant work to this paper is MultiGraspLLM [20], which fine-tunes a VLM to predict dexterous hand pose tokens end-to-end. However, the generated grasp poses are not diverse enough.

3. Notations and Task Specification

We formulate the task of **language-instructed dexterous grasp generation** as follows: The input is a single-view colored point cloud P of the object placed on a table, accompanied by a language instruction T that specifies the semantic object part S_i to be grasped and the grasping style. The details of language instructions are elaborated in §4.4.

The output is a dexterous hand pose that correctly grasps the desired object part with the desired grasping style, as described in the input language instructions. A grasp is represented as $g = (T, R, \theta)$, where $T \in \mathbb{R}^3$ and $R \in \mathrm{SO}(3)$ define the wrist pose, and $\theta \in \mathbb{R}^d$ specifies the joint angles of the hand. We use the Shadow Hand, for which $d = 22$.

4. DexGraspNet 3.0 Dataset

4.1. Dataset Statistics

Table 1 summarizes the key characteristics of the DexGraspNet 3.0 dataset. DexGraspNet 3.0 comprises 170 million dexterous grasps across 174k objects, making it the largest dexterous grasp dataset to date in terms of both grasp pose and object number. Each grasp is validated using the physics-based simulator IsaacGym [29] and paired with semantic captions and part-level annotations, resulting in 170M pose-caption pairs designed for training VLG models. The dataset visualization is shown in Fig. 2.

4.2. Object Preparation and Part Segmentation

Objects are sourced from the Objaverse [9] dataset and filtered using GPT-4o [1], following [34]. The assets are then processed with ManifoldPlus [15] and CoACD [47] to generate collision meshes, yielding 229K valid objects. For each valid object, GPT-4o [1] estimates a reasonable size, and normalization is performed accordingly (see Appendix Sec. B for details). We use SAMesh [38] to perform zero-shot geometry-based part segmentation on colorless collision meshes. Fig. 2 presents visualizations of the segmentation results, providing sufficient functional priors. The part-segmented objects are rendered from multiple views, and part names are automatically labeled using set-of-mark prompting [53] with GPT-4o.

4.3. Part-aware Dexterous Grasp Generation

Our grasp synthesis pipeline is shown in Fig. 3. It is built upon the advanced analytic-based method [6, 57], which uses cuRobo [37] to support massive parallelization on GPUs. To adjust the previous semantic-unaware pipeline for us, we propose a part-aware hand pose initialization strategy and several energy functions, as introduced below.

4.3.1. Part-Aware Hand Pose Initialization

The initial hand pose is regarded as greatly affecting the result of gradient-based optimization, as observed in DexGraspNet [46]. Although that work proposes an initialization method, it is not suitable to our scenario, which requires grasping a specific part. As shown in Fig. 3, we first generate the oriented bounding boxes (OBB) of object parts and sample grasp points from certain areas on the part surface. Then, we set the palm pose and initial joint angles using rules that rely on the geometric cues indicated by the OBB. The wrist pose is further randomly jittered to obtain a diverse distribution. More details are in the Appendix.

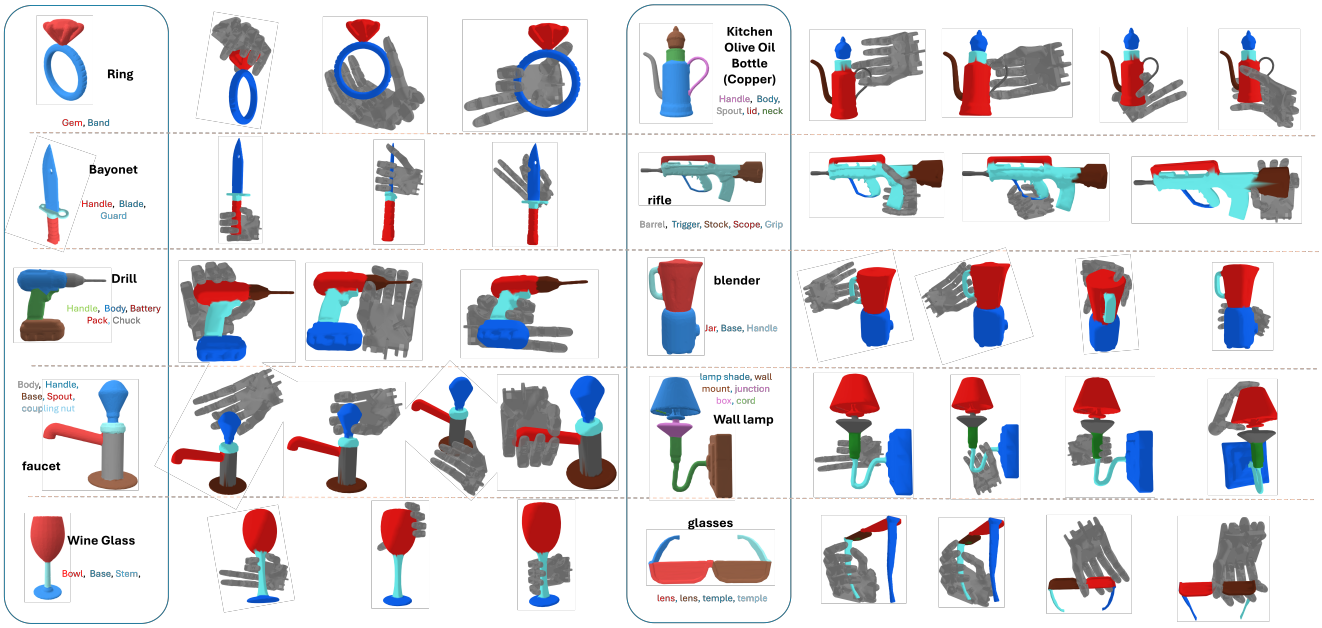

Figure 2. **Visualization of part-aware dexterous grasp poses in DexGraspNet3.0**. The left columns visualize sample objects together with part segmentation generated by SAMesh [38] and captioned by GPT-4o [30]. On the right are part-aligned grasp poses generated by our optimization pipeline. Each grasp makes contact with a single object part and naturally aligns with the way humans grasp objects.

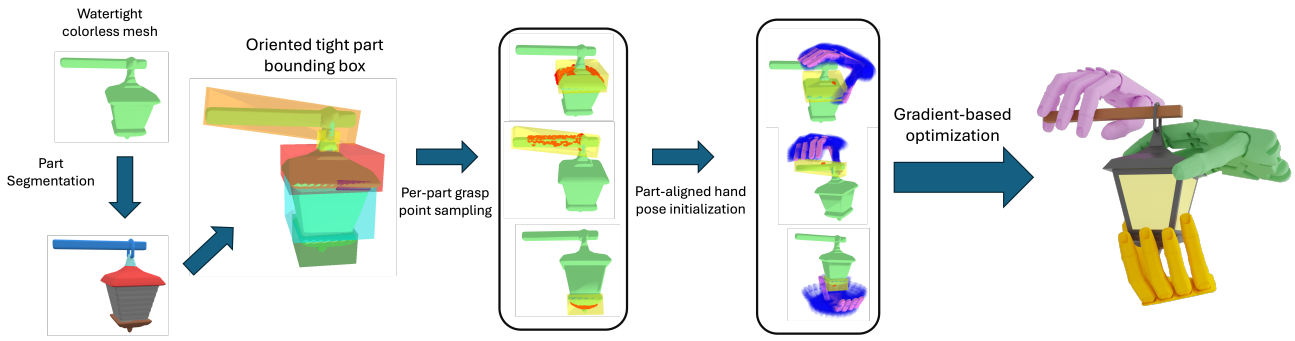

Figure 3. **Grasp pose generation pipeline**. Given a watertight colorless mesh of an object, we perform part segmentation with SAMesh [38] and fit oriented tight bounding boxes for each part. On each part, we sample grasp points on certain areas and align hand initialization poses with the part by leveraging geometric cues in bounding box parameters and grasp points. We further jitter each initial hand pose and run batched gradient-based optimization, which generates final dexterous grasp poses on each object part.

4.3.2. Objectives for Gradient-based Optimization

In this section, we formulate the physics-based energy functions used for gradient-based optimization.

LP-based differentiable force closure energy E_{FC}. Many recent works [6, 19, 25, 57] propose different kinds of differentiable force closure metrics to evaluate the grasp quality. We adopt the variant proposed in DexGraspNet 2.0 [57] to balance the speed and performance. On the one hand, our LP-based energy uses linear programming (LP) to adjust the contact forces, relaxing the assumption of equal contact force in the original DFC metric [25, 46]. On the other hand, our energy assumes no friction, which avoids the heavy computation of quadratic programming as in BODex [6]. More details are in the Appendix.

Part-contact energy To encourage a better contact convergence between the hand and the desired part, We follow the collision detection algorithm in the IPC simulator [22] and define a truncated barrier function E_{bar} that repulses fingertips from object surfaces outside the target part:

$$E_{bar} = \sum_{n=1}^{5} \sum_{j \neq i} b(d(x_n, p_j), d_{thr}) \tag{1}$$

$$b(d, d_{thr}) = \begin{cases} -(d - d_{thr})^2 ln\left(\frac{d}{d_{thr}}\right), & 0 < d < d_{thr} \\ 0, & d \geq d_{thr} \end{cases} \tag{2}$$

Where $\{x_n\}_{n=1}^{5}$ are the fingertips, p_j are point clouds sampled from object surface outside the target part s_i. E_{bar} goes to infinity when any of the fingers make contact with

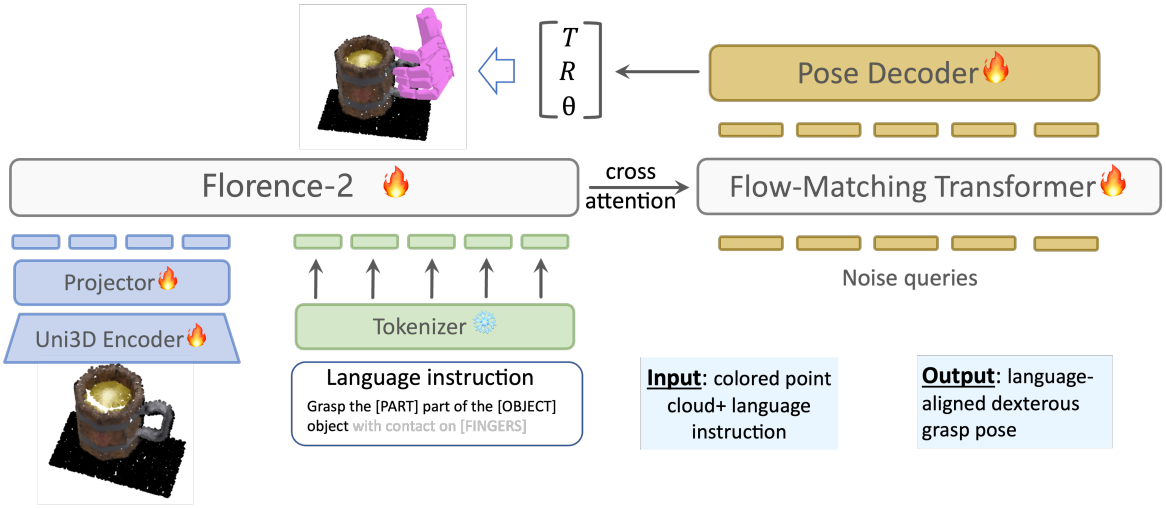

Figure 4. **Pipeline of our DexVLG model.** It first employs Uni3D [62] to encode the colored point cloud, followed by Florence-2 [50] to process and fuse the projected visual features and language tokens. A flow-matching denoise head is conditioned on the VLM output embeddings, while a pose decoder generates the desired grasp pose.

an object outside part s_i, hence strictly enforces part alignment when the stepsize is small enough.

Distance energy We minimize the distance between fingertips and the object to ensure contact and encourage the palm contact points to keep a distance $d_0 = 1cm$ from the object.

$$E_{dis} = \sum_{i=1}^{n} d(x_n, O) + \omega_{palm} |d(x_{palm}, O) - d_0| \quad (3)$$

Besides, we implement several regularization energies, aggregated as E_{reg}, to prevent hand-object collision, hand self-collision and encourage the contact points to align with the front side of fingers. The complete energy function is

$$E = \omega_{FC} E_{FC} + \omega_{bar} E_{bar} + \omega_{dis} E_{dis} + \omega_{reg} E_{reg} \quad (4)$$

4.4. Grasp Validation and Captioning

To obtain a high-quality dataset free of undesirable poses after optimization is complete, we validate all the final poses with a physics-based simulator IsaacGym [29]. We evaluate grasp poses based on four criteria, considering only those that meet all of them as valid: 1) No penetration between the hand and object; 2) No self-penetration within the hand; 3) Stability against gravity in all six axis-aligned directions during simulation; 4) Part alignment, ensuring any hand link in contact with the object is closer to the intended part than to any other part.

We caption each grasp pose with the template **"Grasp the {part} of the {object} object, with contacts on {fingers}"**, where {part} and {object} are the part name and object name inferred by GPT-4o. The part-alignment condition ensures that each grasp pose has a meaningful part name in correspondence. {fingers} lists the names of all fingers in contact with the object part, which is checked in simulation. Each caption contains rich semantic and contact information for the model to learn.

4.5. Table-top Scene Generation and Rendering

The above grasping poses are generated for floating objects, but the object is often placed on a table in the real world. Therefore, we also need to generate diverse poses for objects being stably placed on a table. Following Open6DOR [10], we uniformly sample N=1000 initial rotations from SO(3) and drop the object from a height of 10cm onto the ground. We simulate for 5 seconds and all stabilized poses are collected and deduplicated. Then we transform the grasp poses using the generated object poses and filter out those grasps that have collisions with the table. Each scene is rendered from eight views using a RealSense D415 RGBD camera in Blender. The camera poses are visualized in the Appendix.

5. DexVLG Model

We propose a large vision-language-grasp model, called DexVLG, to tackle the task of language-instructed dexterous grasping. As illustrated in Fig. 4, DexVLG tasks a single-view point cloud observation and a language instruction as input, and outputs a grasp pose that satisfies the language instruction.

5.1. Point Cloud (PC) Encoder

The PC encoder takes colored point clouds from a single view as input. There are lots of foundational PC encoders [11, 31–33] from pretraining. We adopt the pretrained Uni3D [62] backbone, which has a ViT [12]-based architecture scaled from small (23M) to large (307M). Uni3D [62] is pretrained to align point cloud features with

CLIP [36] features via contrastive learning [62], therefore has the ability to extract semantic information from raw point clouds. The point cloud is downsampled into a fixed number $n_p = 10000$ with furthest point sampling before being given to the encoder. The encoded 3D features are then fed into an MLP projector to align the PC features with pre-trained large language models.

5.2. Language Foundation Model

We adopt the LLM base model and language tokenizer from Florence-2 [50], which varies in model parameter size from Base (232M) to Large (771M). We concatenate PC features with language embeddings to create the input for the large language model. Following Transfusion [61] and π_0 [3], the LLM will share the cross-attention with the flow-matching-based pose prediction head.

5.3. Flow Matching-based Grasp Generation

We use the flow matching-based pose denoising module to generate the dexterous grasp pose, which is learned by minimizing the mean square objective:

$$\min_{v} \mathbb{E}_{(t,X_0,X_1)\sim\gamma} || \frac{\mathrm{d}}{\mathrm{d}t} X_t - v(X_t,t) ||^2 \tag{5}$$

where $X_t = \phi(X_0,X_1,t)$ is any time-differentiable interpolation between ground truth grasp pose X_1 and a sample X_0 from noise distribution, with $\frac{\mathrm{d}}{\mathrm{d}t} X_t = \partial_t \phi(X_0,X_1,t)$. The denoising process is conditioned on the LLM's hidden states, with the denoise module sharing transformer architecture with the LLM. An MLP serves as the pose decoder, generating the grasp pose by determining the hand base's translation and rotation along with the joint angles of each finger.

6. Experiments

We evaluate DexVLG against baseline models to answer the following questions:

Does DexVLG zero-shot generate high-quality dexterous grasp poses on a variety of objects and semantic parts? How well do DexVLG-predicted grasp poses align with language instructions? We define metrics that evaluate the quality of grasp poses and how well the poses align with language instruction, and test DexVLG in diverse benchmarks in simulation.

Is it necessary to use large vision-language models to address the task of instruction-aligned dexterous grasp generation? We compare DexVLG against baselines implemented with small model, and analyze the benefits of leveraging large vision-language model.

6.1. Training Details

The entire DexVLG model undergoes single-stage full-parameter fine-tuning, with only the language tokenizer frozen and all other modules updated from end-to-end fine-tuning. The learning rate is set to 6×10^{-5}, with a warm-up strategy applied for the first 3 epochs. Weight decay is 1×10^{-4}. The Adam optimizer is used for training over 230 epochs on 64 NVIDIA RTX 4090 GPUs. Here, we define an epoch as sampling a grasp pose for a single part.

6.2. Evaluation Metrics

The following metrics are used to evaluate the **quality** and **instruction alignment accuracy** of grasp poses generated by different models in simulation.

Simulation Success Rate (Suc) represents the percentage of successful grasp executions in simulation, defined as lifting the object up 3cm from the table and holding for 1s. Our implementation follows the protocol in [57].

Part Touch Accuracy (PTA) represents the percentage of grasp poses that touch the target semantic part, evaluated by checking whether at least one finger in the predicted pose is less than 1cm away from the part and is closer to the desired part than any other part.

Part Grasp Accuracy (PGA) represents the percentage of grasp poses that form a grasping pose at the target part, evaluated by checking whether at least three fingers in the predicted pose are less than 1cm away from the part and are closer to the desired part than any other part.

6.3. Simulation Benchmark Result

We compare DexVLG against DexGraspNet2.0 [46], a diffusion-based small model for dexterous grasp generation in tabletop scenes. We retrain DGN2.0 on DexGraspNet3.0 dataset, dubbed **DGN2.0***. To further equip DGN2.0 with language understanding ability, we concatenate the CLIP [36] embedding of text instructions with point cloud features, and supervise with part-specific ground truth labels. We dub this augmented baseline **DGN2.0*+CLIP**.

Benchmark	Metric	DGN2.0*	DGN2.0* +CLIP	**Ours**
LVIS-Seen	**Suc**↑	70.8	67.8	**87.7**
	PTA↑	54.0	55.2	**70.7**
	PGA↑	38.5	40.5	**62.1**
Unseen	**Suc**↑	57.7	56.0	**79.1**
	PTA↑	63.9	64.7	**68.2**
	PGA↑	25.3	26.1	**36.3**
SamPart3D	**Suc**↑	56.8	55.1	**76.3**
	PTA↑	49.2	50.8	**66.0**
	PGA↑	38.8	39.4	**52.0**

Table 2. **Result on simulation benchmarks**.

To evaluate the models, we curate three benchmarks in simulation. In each benchmark, we manually select well-

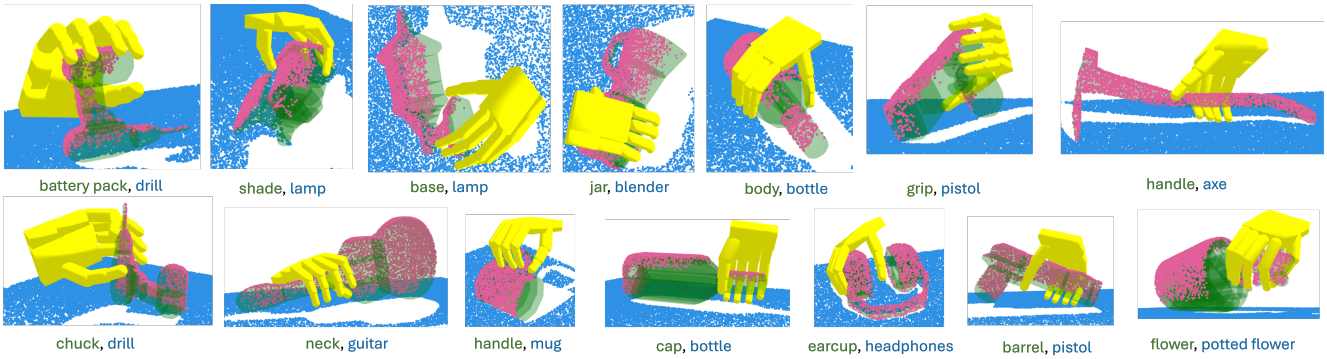

Figure 5. **Visualization of grasp poses predicted by DexVLG in simulation.** The object mesh is drawn only for visualization. The model input is a single-view point cloud, the color present in this figure is painted only for visualization. Each grasp is instructed with "Grasp the [PART] part of [OBJECT] object".

Method	Suc↑	Part Grasp↑	CMA↑
DGN2.0*+CLIP	68.2	35.2	21.8
+ contact label	72.3	33.8	23.0
Ours	73.0	33.9	27.0
+ contact label	**76.1**	**48.1**	**29.8**

Table 3. **Result of contact mode learning on LVIS-Seen benchmark in simulation**. The **contact mode accuracy (CMA)** metric refers to contact mode accuracy, which reflects the rate of grasp poses that match the instructed contact mode, allowing differences with at most one finger contact.

segmented objects and filter their poses on table such that the target parts for grasping are not occluded by the table.

- The **LVIS-Seen** benchmark consists of 40 seen objects in the Objaverse-LVIS split.
- The **Unseen** benchmark consists of 84 Objaverse objects unseen in the training process.
- The **SamPart3D** benchmark consists of 56 Objaverse objects segmented and semantically annotated by SamPart3D [55] using methods different from our work.

6.3.1. Part-conditioned Grasp Generation

In this experiment, we train and infer with input language instruction "**Grasp the {part} part of the {object} object**". The quantitative results are shown in Tab. 2. Our DexVLG outperforms DGN2.0+CLIP in terms of both simulation success rate and part accuracy in all benchmarks, demonstrating superior performance. DexVLG robustly generalizes to grasping unseen objects and retains 79% success rate and 36% part grasp accuracy. It also generalizes with respect to part segmentation methods and retains a 76% success rate and a 52% part grasp accuracy in the SamPart3D benchmark. The qualitative results are shown in Fig. 5.

It is worth noting that even though the Florence-2 LLM backbone [50] is not pretrained on object part understanding tasks and haven't learned about the novel part names of objects in **Unseen** benchmark through finetuning, the DexVLG model still learns to follow instruction of grasping these parts and achieve substantially higher part grasp accuracy than baselines. This result demonstrates the strong generalizability of VLMs in learning language-aligned dexterous grasping tasks.

Interestingly, we find that augmenting the DGN2.0 baseline with CLIP features enhances its language alignment ability at the cost of harming the quality of generated poses, reflected in a drop in simulation success rate. This trade-off behavior indicates **small models are limited in capacity, which hinders them from learning complex tasks such as generating instruction-aligned dexterous grasps.** On the other hand, Tab. 4 shows training with language condition enhances the performance of DexVLG, which indicates **the large capacity of VLM is necessary for learning instruction-aligned dexterous grasp generation.**

6.3.2. Contact Mode-conditioned Grasp Generation

In this experiment, we benchmark different models on a more complex instruction following task: inference with language instruction "**Grasp the {part} part of the {object} object with contact on {fingers}**", where {fingers} are the names of fingers that we want to make contact with the object. We compare models trained with **(dubbed "+contact label")** and without contact mode labels on the **LVIS-Seen** benchmark. The quantitative results are shown in Tab. 3. Models trained without a contact mode label are confused by this extra instruction and degenerate in performance. On the other hand, when trained with contact mode annotation, DexVLG effectively learn to follow the extra instruction and improves performance. This experiment demonstrates **large VLMs are capable of learning more complex instruction-following grasp generation tasks, which small model struggles to learn**.

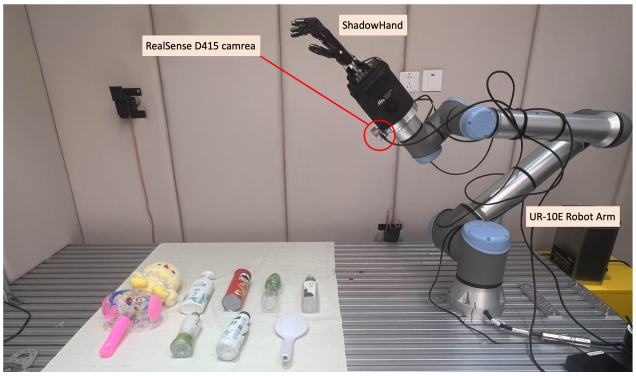

Figure 6. **real-world experiment setting**.

6.4. Real-World Experiments

Our real-world experiments utilize a ShadowHand mounted on UR10e robotic arm. An Intel RealSense D415 camera is mounted at the wrist and captures static single-view colored point cloud from a 45-degree lookdown perspective. During the experiment, we prompt DexVLG with language instruction specifying desired parts, such as "grasp the neck of the bottle". The generated poses are filtered against safety constraints and executed with motion planning. We report **Grasp Success Rate** as the rate of successfully lifting up, and report **Part Accuracy** by human examination. DexVLG achieves 80% success rate and 75% part accuracy with these simple objects. Limited by hardware workspace and safety concerns, we do not report real-world results with more complex objects.

6.5. Ablation Study

We conduct ablation studies on model architecture, data scale, and input information. As shown in Table 5, training data scale plays a crucial role in performance, with a 10% to 100% increase significantly improving success rate (Suc) and part grasp accuracy (PGA) across all datasets. The most substantial gains occur in unseen objects and 3D part recognition, emphasizing the importance of larger datasets for better generalization. Our model size study (Table 6) shows that increasing parameters from 255M to 1B provides only marginal improvements in success rate, with inconsistent effects on part accuracy. The 320M model performs best for part accuracy on LVIS-Seen and Unseen, while the 1B model excels in overall success rate, especially for SamPart3D. This suggests that larger models do not always enhance part-level understanding, and a mid-sized model (320M) balances efficiency and performance. Our input information study (Table 7) reveals that adding color to point clouds (PC w/ color) significantly improves both success rate and part grasp accuracy. The biggest gains appear in LVIS-Seen (Suc: +27.7, Part G: +23.5) and SamPart3D (Suc: +19.8, Part G: +13.1). These findings highlight the critical role of color in enhancing both object-level and part-level understanding in 3D.

Data	LVIS-Seen		Unseen		SamPart3D	
	Suc↑	**PGA↑**	**Suc↑**	**PGA↑**	**Suc↑**	**PGA↑**
with instruction	87.7	62.1	79.1	36.3	76.3	52.0
no instruction	84.4	–	64.5	–	70.2	–

Table 4. **Ablation study for input language instruction**.

Data	LVIS-Seen		Unseen		SamPart3D	
	Suc↑	**PGA↑**	**Suc↑**	**PGA↑**	**Suc↑**	**PGA↑**
10%	49.7	12.5	32.3	7.9	28.4	11.7
20%	75.3	39.1	54.0	18.3	53.4	27.0
100%	87.7	62.1	79.1	36.6	76.3	52.0

Table 5. **Ablation study on training data scaling**. We explore the data efficiency from 10% (about 17k objects) to 100%.

Param	LVIS-Seen		Unseen		SamPart3D	
	Suc↑	**PGA↑**	**Suc↑**	**PGA↑**	**Suc↑**	**PGA↑**
255M	74.8	35.6	55.2	16.4	47.6	26.5
320M	72.3	41.6	53.7	20.5	50.0	22.8
1B	75.3	39.1	54.0	18.3	53.4	27.0

Table 6. **Ablation study for model size**. The 225M model uses Florence-2-base and Uni3D-small, 320M for Florence-2-base and Uni3D-base, and 1B for Florence-2-large and Uni3D-large.

Input	LVIS-Seen		Unseen		SamPart3D	
	Suc↑	**PGA↑**	**Suc↑**	**PGA↑**	**Suc↑**	**PGA↑**
PC w/o color	47.6	15.6	35.0	10.4	33.6	13.9
PC w/ color	75.3	39.1	54.0	18.3	53.4	27.0

Table 7. **Ablation study on input information**. We compare the impact of using colored point clouds.

7. Limitations and Conclusions

In this paper, we present DexVLG, an end-to-end language-aligned dexterous grasp generation model that leverages the capacity of large VLMs, trained with our synthesized large-scale DexGraspNet3.0 dataset. DexVLG achieves state-of-the-art performance in both grasp success and part accuracy in simulation, and achieves 80% success rate in grasping simple objects in the real world. Nonetheless, our work has several limitations. As training poses in the DexGraspNet3.0 dataset is synthesized with floating hands without considering the hand-arm workspace, many poses sampled by DexVLG are unsafe to execute in real-world. As another limitation, our method does not support effective ranking of generated grasp poses. Ranking large batches of samples by likelihood score [13], as done in [57], is infeasible for VLM-based models as retaining gradients with respect to VLM parameters costs huge GPU memory. We leave developing effective sample ranking methods as future work.

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. 3

[2] Ananye Agarwal, Shagun Uppal, Kenneth Shaw, and Deepak Pathak. Dexterous functional grasping. In *7th Annual Conference on Robot Learning*, 2023. 3

[3] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024. 2, 6

[4] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023. 3

[5] Jiayi Chen, Yuxing Chen, Jialiang Zhang, and He Wang. Task-oriented dexterous grasp synthesis via differentiable grasp wrench boundary estimator. *arXiv preprint arXiv:2309.13586*, 2023. 2, 3

[6] Jiayi Chen, Yubin Ke, and He Wang. Bodex: Scalable and efficient robotic dexterous grasp synthesis using bilevel optimization. *arXiv preprint arXiv:2412.16490*, 2024. 2, 3, 4

[7] Sirui Chen, Jeannette Bohg, and C Karen Liu. Springgrasp: An optimization pipeline for robust and compliant dexterous pre-grasp synthesis. *arXiv preprint arXiv:2404.13532*, 2024. 2

[8] Zoey Qiuyu Chen, Karl Van Wyk, Yu-Wei Chao, Wei Yang, Arsalan Mousavian, Abhishek Gupta, and Dieter Fox. Learning robust real-world dexterous grasping policies via implicit shape augmentation. *arXiv preprint arXiv:2210.13638*, 2022. 2

[9] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 2, 3

[10] Yufei Ding, Haoran Geng, Chaoyi Xu, Xiaomeng Fang, Jiazhao Zhang, Songlin Wei, Qiyu Dai, Zhizheng Zhang, and He Wang. Open6dor: Benchmarking open-instruction 6-dof object rearrangement and a vlm-based approach. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7359–7366. IEEE, 2024. 5

[11] Runpei Dong, Zekun Qi, Linfeng Zhang, Junbo Zhang, Jianjian Sun, Zheng Ge, Li Yi, and Kaisheng Ma. Autoencoders as cross-modal teachers: Can pretrained 2d image transformers help 3d representation learning? In *The Eleventh International Conference on Learning Representations*, 2023. 5

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 5

[13] Will Grathwohl, Ricky T. Q. Chen, Jesse Bettencourt, Ilya Sutskever, and David Duvenaud. Ffjord: Free-form continuous dynamics for scalable reversible generative models, 2018. 8

[14] Jinglue Hang, Xiangbo Lin, Tianqiang Zhu, Xuanheng Li, Rina Wu, Xiaohong Ma, and Yi Sun. Dexfuncgrasp: A robotic dexterous functional grasp dataset constructed from a cost-effective real-simulation annotation system. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10306–10313, 2024. 2, 3

[15] Jingwei Huang, Yichao Zhou, and Leonidas Guibas. Manifoldplus: A robust and scalable watertight manifold surface generation method for triangle soups. *arXiv preprint arXiv:2005.11621*, 2020. 3

[16] Linyi Huang, Hui Zhang, Zijian Wu, Sammy Christen, and Jie Song. Fungrasp: Functional grasping for diverse dexterous hands. *arXiv preprint arXiv:2411.16755*, 2024. 2, 3

[17] 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

[18] 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. 2, 3

[19] Albert H Li, Preston Culbertson, Joel W Burdick, and Aaron D Ames. Frogger: Fast robust grasp generation via the min-weight metric. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6809–6816. IEEE, 2023. 2, 4

[20] Haosheng Li, Weixin Mao, Weipeng Deng, Chenyu Meng, Haoqiang Fan, Tiancai Wang, Ping Tan, Hongan Wang, and Xiaoming Deng. Multi-grasplm: A multimodal llm for multi-hand semantic guided grasp generation. *arXiv preprint arXiv:2412.08468*, 2024. 3

[21] Kailin Li, Jingbo Wang, Lixin Yang, Cewu Lu, and Bo Dai. Semgrasp: Semantic grasp generation via language aligned discretization. In *European Conference on Computer Vision*, pages 109–127. Springer, 2025. 3

[22] Minchen Li, Zachary Ferguson, Teseo Schneider, Timothy R Langlois, Denis Zorin, Daniele Panozzo, Chenfanfu Jiang, and Danny M Kaufman. Incremental potential contact: intersection-and inversion-free, large-deformation dynamics. *ACM Trans. Graph.*, 39(4):49, 2020. 4

[23] Fangchen Liu, Kuan Fang, Pieter Abbeel, and Sergey Levine. Moka: Open-vocabulary robotic manipulation through mark-based visual prompting. In *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*, 2024. 3

[24] Minghua Liu, Yinhao Zhu, Hong Cai, Shizhong Han, Zhan Ling, Fatih Porikli, and Hao Su. Partslip: Low-shot part segmentation for 3d point clouds via pretrained image-

language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21736–21746, 2023. 2

[25] 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, 4

[26] Xueyi Liu, Jianibieke Adalibieke, Qianwei Han, Yuzhe Qin, and Li Yi. Dextrack: Towards generalizable neural tracking control for dexterous manipulation from human references. *arXiv preprint arXiv:2502.09614*, 2025. 2

[27] Marios Loizou, Siddhant Garg, Dmitry Petrov, Melinos Averkiou, and Evangelos Kalogerakis. Cross-shape attention for part segmentation of 3d point clouds. In *Computer Graphics Forum*, page e14909. Wiley Online Library, 2023. 2

[28] Ziqi Ma, Yisong Yue, and Georgia Gkioxari. Find any part in 3d. *arXiv preprint arXiv:2411.13550*, 2024. 2

[29] 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 *NeurIPS*, 2021. 2, 3, 5

[30] OpenAI. Introducing gpt-4o and more tools to chatgpt free users. 2024. 2, 4

[31] Zekun Qi, Runpei Dong, Guofan Fan, Zheng Ge, Xiangyu Zhang, Kaisheng Ma, and Li Yi. Contrast with reconstruct: Contrastive 3d representation learning guided by generative pretraining. In *International Conference on Machine Learning*, pages 28223–28243. PMLR, 2023. 5

[32] Zekun Qi, Muzhou Yu, Runpei Dong, and Kaisheng Ma. Vpp: Efficient conditional 3d generation via voxel-point progressive representation. *Advances in Neural Information Processing Systems*, 36:26744–26763, 2023.

[33] Zekun Qi, Runpei Dong, Shaochen Zhang, Haoran Geng, Chunrui Han, Zheng Ge, Li Yi, and Kaisheng Ma. Shapellm: Universal 3d object understanding for embodied interaction. In *European Conference on Computer Vision*, pages 214–238. Springer, 2025. 5

[34] Zekun Qi, Wenyao Zhang, Yufei Ding, Runpei Dong, Xinqiang Yu, Jingwen Li, Lingyun Xu, Baoyu Li, Xialin He, Guofan Fan, Jiazhao Zhang, Jiawei He, Jiayuan Gu, Xin Jin, Kaisheng Ma, Zhizheng Zhang, He Wang, and Li Yi. So-far: Language-grounded orientation bridges spatial reasoning and object manipulation. *CoRR*, abs/2502.13143, 2025. 3

[35] Yuzhe Qin, Wei Yang, Binghao Huang, Karl Van Wyk, Hao Su, Xiaolong Wang, Yu-Wei Chao, and Dieter Fox. Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system. *arXiv preprint arXiv:2307.04577*, 2023. 2

[36] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 6

[37] Balakumar Sundaralingam, Siva Kumar Sastry Hari, Adam Fishman, Caelan Garrett, Karl Van Wyk, Valts Blukis, Alexander Millane, Helen Oleynikova, Ankur Handa, Fabio Ramos, et al. Curobo: Parallelized collision-free robot motion generation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8112–8119. IEEE, 2023. 3

[38] George Tang, William Zhao, Logan Ford, David Benhaim, and Paul Zhang. Segment any mesh: Zero-shot mesh part segmentation via lifting segment anything 2 to 3d. *arXiv preprint arXiv:2408.13679*, 2024. 2, 3, 4

[39] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 2

[40] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024. 2, 3

[41] 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. 2

[42] Dylan Turpin, Tao Zhong, Shutong Zhang, Guanglei Zhu, Eric Heiden, Miles Macklin, Stavros Tsogkas, Sven Dickinson, and Animesh Garg. Fast-grasp'd: Dexterous multi-finger grasp generation through differentiable simulation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8082–8089. IEEE, 2023. 2, 3

[43] Weikang Wan, Haoran Geng, Yun Liu, Zikang Shan, Yaodong Yang, Li Yi, and He Wang. Unidexgrasp++: Improving dexterous grasping policy learning via geometry-aware curriculum and iterative generalist-specialist learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3891–3902, 2023. 2

[44] Beichen Wang, Juexiao Zhang, Shuwen Dong, Irving Fang, and Chen Feng. Vlm see, robot do: Human demo video to robot action plan via vision language model. *arXiv preprint arXiv:2410.08792*, 2024. 3

[45] Chen Wang, Haochen Shi, Weizhuo Wang, Ruohan Zhang, Li Fei-Fei, and C Karen Liu. Dexcap: Scalable and portable mocap data collection system for dexterous manipulation. *arXiv preprint arXiv:2403.07788*, 2024. 2

[46] 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. 2, 3, 4, 6

[47] Xinyue Wei, Minghua Liu, Zhan Ling, and Hao Su. Approximate convex decomposition for 3d meshes with collision-aware concavity and tree search. *ACM Transactions on Graphics (TOG)*, 41(4):1–18, 2022. 3

[48] Yi-Lin Wei, Jian-Jian Jiang, Chengyi Xing, Xian-Tuo 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. 2, 3

[49] Rina Wu, Tianqiang Zhu, Xiangbo Lin, and Yi Sun. Cross-category functional grasp transfer. *IEEE Robotics and Automation Letters*, 2024. 2, 3

[50] Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4818–4829, 2024. 5, 6, 7

[51] Guo-Hao Xu, Yi-Lin Wei, Dian Zheng, Xiao-Ming Wu, and Wei-Shi Zheng. Dexterous grasp transformer, 2024. 2

[52] Yinzhen Xu, Weikang Wan, Jialiang Zhang, Haoran Liu, Zikang Shan, Hao Shen, Ruicheng Wang, Haoran Geng, Yijia Weng, Jiayi Chen, et al. Unidexgrasp: Universal robotic dexterous grasping via learning diverse proposal generation and goal-conditioned policy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4737–4746, 2023. 2

[53] Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023. 3

[54] 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, 3

[55] Yunhan Yang, Yukun Huang, Yuan-Chen Guo, Liangjun Lu, Xiaoyang Wu, Edmund Y Lam, Yan-Pei Cao, and Xihui Liu. Sampart3d: Segment any part in 3d objects. *arXiv preprint arXiv:2411.07184*, 2024. 2, 7

[56] Hui Zhang, Sammy Christen, Zicong Fan, Otmar Hilliges, and Jie Song. Graspxl: Generating grasping motions for diverse objects at scale. In *European Conference on Computer Vision*, pages 386–403. Springer, 2024. 2

[57] Jialiang Zhang, Haoran Liu, Danshi Li, XinQiang Yu, Haoran Geng, Yufei Ding, Jiayi Chen, and He Wang. Dexgraspnet 2.0: Learning generative dexterous grasping in large-scale synthetic cluttered scenes. In *8th Annual Conference on Robot Learning*, 2024. 2, 3, 4, 6, 8

[58] Wenyao Zhang, Hongsi Liu, Zekun Qi, Yunnan Wang, XinQiang Yu, Jiazhao Zhang, Runpei Dong, Jiawei He, He Wang, Zhizheng Zhang, et al. Dreamvla: A vision-language-action model dreamed with comprehensive world knowledge. *arXiv preprint arXiv:2507.04447*, 2025. 3

[59] 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 method for multi-dexterous robotic hands. *arXiv preprint arXiv:2407.09899*, 2024. 2

[60] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16259–16268, 2021. 2

[61] Chunting Zhou, LILI YU, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. In *The Thirteenth International Conference on Learning Representations*, 2025. 6

[62] Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3d: Exploring unified 3d representation at scale. In *The Twelfth International Conference on Learning Representations*, 2024. 5, 6

[63] Yuchen Zhou, Jiayuan Gu, Xuanlin Li, Minghua Liu, Yunhao Fang, and Hao Su. Partslip++: Enhancing low-shot 3d part segmentation via multi-view instance segmentation and maximum likelihood estimation. *arXiv preprint arXiv:2312.03015*, 2023. 2