IFG: Internet-Scale Guidance for Functional Grasping Generation

Anonymous Review

Fig. 1: IFG enables the generation of dexterous, functional grasps in cluttered, realistic scenes. It first uses a vision-language model to identify task-relevant regions on objects, then uses geometrically precise force closure in simulation to ground the finger joints. The resulting dataset, and the diffusion model trained on it, encode both semantic and geometric understanding of the scene without any hand-collected data.

***Abstract*—Large Vision Models trained on internet-scale data have demonstrated strong capabilities in segmenting and semantically understanding object parts, even in cluttered, crowded scenes. However, while these models can direct a robot toward the general region of an object, they lack the geometric understanding required to precisely control dexterous robotic hands for 3D grasping. To overcome this, our key insight is to leverage simulation with a force-closure grasping generation pipeline that understands local geometries of the hand and object in the scene. Because this pipeline is slow and requires ground-truth observations, the resulting data is distilled into a diffusion model that operates in real-time on camera point clouds. By combining the global semantic understanding of internet-scale models with the geometric precision of a simulation-based locally-aware force-closure, IFG achieves high-performance semantic grasping without any manually collected training data.**

I. INTRODUCTION

Recent advances in vision-language models (VLMs) have led to impressive results across a range of perception tasks, including image captioning, visual question answering, and open-world object recognition. Trained on large-scale datasets pairing images with natural language, these models exhibit a strong ability to align visual and linguistic information, enabling semantic understanding that generalizes across diverse contexts. This success has inspired interest in leveraging VLMs for robotics applications such as instruction following, semantic goal specification, and high-level planning.

While these initial applications show promise, significant limitations remain. Most notably, current VLMs lack a grounded understanding of physical space—they cannot reason about 3D geometry, spatial relationships, or the dynamics of physical interaction. Consequently, they struggle with planning or executing precise motor actions in the real world. Although VLMs can identify and describe visual content, they do not inherently understand how to interact with it. This disconnect between perception and control poses a major challenge in robotic systems.

To make robot learning more scalable and generalizable, we seek an approach that avoids manual data collection methods like teleoperation while enabling zero-shot generalization. A promising direction involves synthetic grasp generation frameworks, which produce large datasets of grasp poses through an optimization process guided by energy functions that approximate force closure, along with evaluation pipelines in simulation. These datasets are often used to train diffusion-based grasp samplers. However, a significant portion of the generated grasps are physically implausible or unnatural. Because grasp proposals are initialized by sampling points around the object's convex hull, many grasps target physically inaccessible or unsuitable regions, reducing the overall efficiency of the generation process.

Moreover, downstream manipulation tasks require the hand to interact with specific, task-relevant regions of objects—what we refer to as useful regions. Existing synthetic pipelines, however, generate grasps indiscriminately over the object surface, leading to noisy datasets that are poorly aligned with the needs of task-conditioned manipulation.

Our approach addresses this gap by combining the high-level semantic understanding of VLMs with physically grounded,

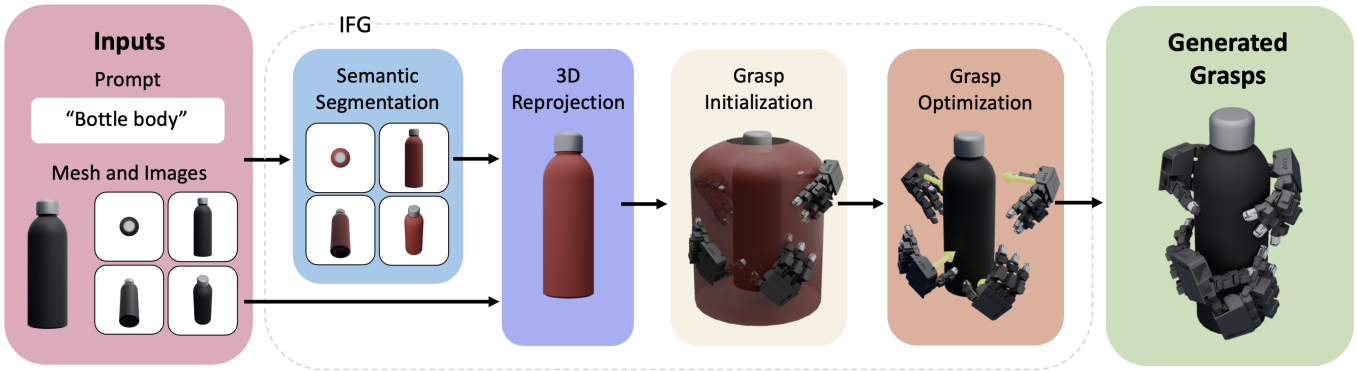

Fig. 2: IFG takes an object mesh and a task prompt as input. To incorporate semantic understanding, it renders the object from multiple viewpoints, applies a VLM-based segmentation model combining SAM[1] and VLPart[**?**], and reprojects the results into 3D space to identify task-relevant regions. For geometric grounding, it initializes a force closure objective at these regions and optimizes for functional grasps. The resulting data is then used to train a diffusion model for fast grasp synthesis from depth.

task-aware representations. We propose a pipeline that translates semantic input specifying a task into predictions of useful regions on objects. By integrating this into the grasp generation process, we enable semantic-guided grasp synthesis, producing stable, natural grasps aligned with the demands of the task. Our pipeline is highly parallelizable, efficient, and compatible with arbitrary objects, scenes (including cluttered environments), and dexterous hands. It generates meaningful grasps without any laborious collection of teleoperation or video data.

II. RELATED WORKS

A. Dexterous Grasping Generation in Simulation

GraspIt! adapts Eigengrasps from a manual database for fast grasp planning, but it doesn't generalize to any object category [3]. Similarly, [4] also uses shape augmentations. Grasp'D uses differentiable contact simulations to create grasps that surround the object by using SDF. [5, 6, 7] DexGraspNet and its follow-up initialize the hand poses in the convex hull of the object and then optimize using the force closure objective.[8, 9]. We use this objective function to generate grasps as well in IFG.

Many grasping pipelines use the data generated to train neural network models. These models can aid in faster generation, the removal of privileged information, or enable generalization. Many use a VAE model [10] to enable this faster generation [9, 11, 12, 13, 14]. Other newer works use the more powerful diffusion model [15] to improve results. [2] To remove the reliance on privileged geometry, point-cloud conditioning with Pointnet [16, 17] can be used. [18] Some use a NeRF [14]. Finally, some use quick test time adaptation to improve the grasping quality past the learned model [19, 11].

B. Vision-based Dexterous Grasping

Instead of generating grasps in simulation there are many datasets that have human hand and object interaction in them. Some datasets have ground truth data using motion capture devices [20, 21] but are more limited in size and variety of grasps [22, 23]. Adding contact information between the object and the hand can help with fine-grained control. [24, 25, 26] While there are larger datasets available, [27, 28], they do not have ground truth hand and object poses. This means that vision methods must be used to extract this, which works in varying accuracy. [29, 30, 31] Once extracted, these grasps can be used in robot hand systems.

A wide range of recent studies have tapped into large-scale human activity datasets to improve various aspects of robot learning. Some focus on deriving cost functions from human behavior [32, 33, 34, 35], while others map human and robot actions to one another [36, 37], whether through aligned demonstrations [38, 39, 40], unaligned examples [41], or direct action correspondences [42]. In addition, the inherent structure of certain datasets—such as those involving tool use [43], or temporal sequences of hand-object interactions [44, 45]—has been used to infer actions or detect salient features like keypoints.

C. VLMs for Robotic Grasping

Recent works have explored integrating large-scale models with robotic grasping, particularly for two-finger grippers. For example, [46] and [47] extract affordances and constraints from LLMs and VLMs to build 3D value maps, which are then used by motion planners to synthesize trajectories in a zero-shot manner. Similarly, [48] incorporates large-scale models but relies on simulation to train downstream policies. Other approaches [49, 50, 51] generate vision-language-action (VLA) representations or language-based plans that can be executed on robotic systems.

III. METHOD

The goal of IFG is to learn a general-purpose dexterous grasping affordance model that takes as input a scene point cloud and a text prompt specifying the object to grasp, and outputs a feasible grasp for a robot hand. To enable this, we must first generate a large grasping dataset with geometrically accurate and semantically meaningful grasps. As shown in Figure 2, given an object mesh and a task prompt, our data generation pipeline identifies task-relevant regions by rendering the object from multiple viewpoints, applying a VLM-based segmentation model, and reprojecting the results into 3D. These

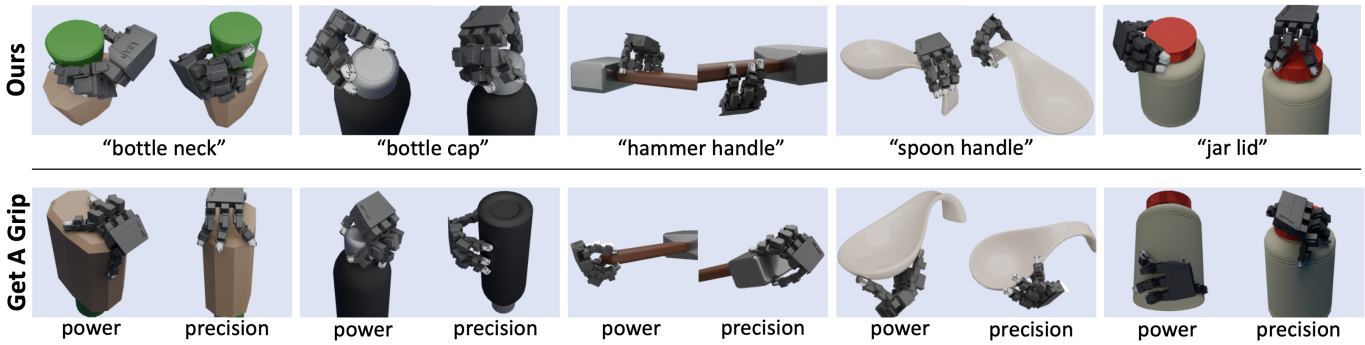

Fig. 3: Compared to Get a Grip's synthetic grasp generation method, our method produces more human-like grasps. For instance, Get a Grip often grasp on the bottom of the bottle, while our method knows to go for the neck. This makes our grasps better for real world scenarios.

semantic regions then guide a grasp optimization process that enforces both stability and functional relevance. The resulting diverse set of robust grasps is distilled into a diffusion model that predicts executable grasps directly from depth input, enabling fast and deployable grasp synthesis in real-world scenarios. We present the pseudocode of our method in Algorithm 1.

A. Dexterous grasp Formulation

We formulate dexterous grasps as follows. A dexterous grasp g is defined as $g = (T, R, \theta)$, where $T \in \mathbb{R}^3$ and $R \in SO(3)$ represent the translation and rotation of the wrist pose, and $\theta \in \mathbb{R}^{\text{DoF}}$ denotes the joint angles of the hand (DoF $= 16$ for LEAP Hand [**?**]).

B. Useful region proposal

IFG leverages knowledge from a VLM f to identify objects of interest and part-level useful regions. To extract 2D semantic knowledge to 3D scenes, we also use a language-conditioned segmentation model g to isolate the object and a part-level segmentation model h to identify regions of interest on the object. Given an object mesh O with k faces $F = \{f^{(1)}, ..., f_{(k)}\}$, we take a set of n RGB images

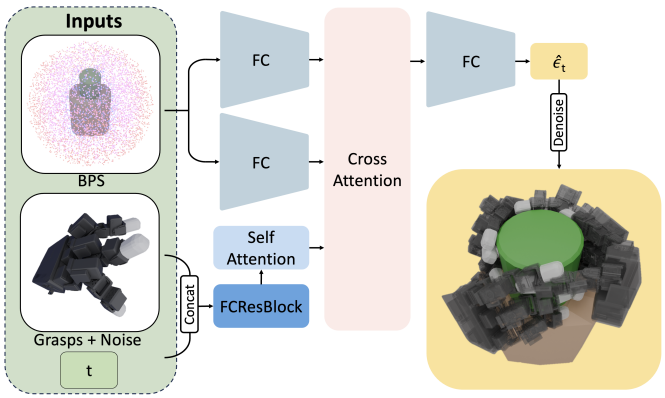

Fig. 4: To enable real-world deployment, the generated grasp data is distilled into a diffusion model. This model is conditioned on a Basis Point Set (BPS) computed from depth camera data, along with a noisy grasp input. Through the denoising process, the model produces refined grasps on the object. The architecture of the diffusion model follows a similar design to DexDiffuser [2].

$V = \{v^{(1)}, \ldots, v^{(n)}\}$ from angles uniformly sampled on a camera initialization surface S. For single object settings S is a spherical surface, and for clustered scenes it is the dome segmented from a sphere to avoid visual occlusion. The VLM is prompted with V to produce m semantic labels of useful regions of O denoted as $f(V) = R = \{r_1, \ldots, r_m\}$. For each label $p_i \in P$, g and h together produces part-level segmentation masks for each image in V conditioned on r_i, represented as $h \circ g(r_i) = S_i = \{s_i^{(1)}, \ldots, s_i^{(n)}\}$, where $s_i^{(j)}$ segments the regions of $v^{(m)}$ that belong to r_i. We use SAM[1] as the object segmentation model and VLPart[**?**] as the part-level segmentation model.

The 2D segmentation masks S_i are deprojected back to 3D points on the object mesh $P_i = \{p_i^{(1)}, \ldots, p_i^{(n)}\}$. However, from certain camera angles a part may be occluded, leading to incorrect segmentation. To address this, P_i further undergoes heuristic-based filtering. A two-means based clustering process where each p_i is assigned to one of the two groups based on the size of their segmentation mask s_i in terms of masked pixel number to filter out unsuccessful segmentation masks. The larger group from the clustering process $\hat{P}_i$ is considered the valid deprojected points. Each point is then associated with the closest face on the object mesh to produce a tally of face counts $T_i = \{t_i^{(1)}, ..., t_i^{(k)}\}$, where $t_i^{(j)} \in \mathbb{N}$. A voting algorithm selects the 60% top faces of the object mesh as the useful region of the object used for the next stage, which we refer to as U.

C. Geometric Grasp synthesis

We compute the segmented convex hull of the object to include only faces projected from U. For each grasp, the hand is initialized on the inflated convex hull by farthest point sampling with random noise added to the pose and finger joint angles. An optimization process performs gradient descent against an energy term

$$E = E_{\text{fc}} + w_{\text{dis}} E_{\text{dis}} + w_{\text{joints}} E_{\text{joints}} + w_{\text{pen}} E_{\text{pen}} + w_{\text{spen}} E_{\text{spen}}$$

where E_{fc} approximates force closure of the grasp, E_{dis} encourages hand-object proximity, based on the contact points of the hand, E_{joints}, E_{pen}, and E_{spen} respectively penalizes joint violations, hand-object penetration, and self-penetration of the hand. For the single object setting, we exclude the tabletop by

Algorithm 1 IFG

Require: VLM f, segmentation models g, h, object mesh O with faces $F = \{f^{(1)}, \ldots, f^{(k)}\}$, n views $V = \{v^{(1)}, \ldots, v^{(n)}\}$ from camera surface S

Semantic Segmentation Region Extraction

Query VLM: $R = f(V) = \{r_1, \ldots, r_m\}$ semantic labels
for each label $r_i \in R$ **do**
  Obtain masks $S_i = h \circ g(r_i) = \{s_i^{(1)}, ..., s_i^{(n)}\}$
  Deproject masks: $P_i = \{p_i^{(1)}, ..., p_i^{(n)}\}$
  Filter P_i with two-means clustering by mask size
  Map filtered points $\hat{P}_i$ to nearest faces, tally counts T_i
end for
Select top 60% faces $U \subseteq F$ as useful regions

Geometric Grasp Synthesis

Build convex hull from U; inflate for sampling
for each grasp initialization **do**
  Place hand on hull via farthest point sampling + random noise
  Optimize energy

$$E = E_{\text{fc}} + w_{\text{dis}} E_{\text{dis}} + w_{\text{joints}} E_{\text{joints}} + w_{\text{pen}} E_{\text{pen}} + w_{\text{spen}} E_{\text{spen}}$$

  Obtain candidate grasp
end for

Simulation Evaluation

for each grasp **do**
  Generate d perturbed grasps by varying joint angles
  Simulate Lift / Pick&Shake tasks in IsaacGym
  Assign smooth label as mean success over $d + 1$ trials
end for
Filter grasps with low success $\rightarrow G$

setting $w_{\text{spen}} = 0$ to produce more diverse grasps. This pipeline is similar to Get a Grip's synthetic pipeline except for a few key modifications. Instead of using precision grasps, which sample contact points only on the fingertips of the hand, we use power grasps by sampling over the inside regions of all fingers, which produce more stable grasps and thus yield a higher success rate. For each grasp, we initialize hand positions on the segmented convex hull instead of the entire hull.

D. Simulation Evaluation

To ensure the robustness of generated grasps, we perform tasks with them in a simulation environment. Each evaluation proceeds in three phases: (1) the grasp and object are initialized in a simulation environment, (2) fingers are closed to secure the object, and (3) task execution is performed. Following Get a Grip, we use a smooth label for each grasp by applying slight perturbations on the finger joint angles to produce d associating grasps, all of which are evaluated in simulation. The hard success rates of all $d + 1$ grasps are averaged to produce the smooth label for the grasp. Grasps with low success rates are filtered out to produce a dataset G of robust, force-closure power grasps. For our experiments, $d = 5$.

Object	GET A GRIP	OURS
water bottle	49.1	**62.8**
large detergent bottle	51.2	**62.5**
spray bottle	43.1	**54.5**
pan	48.1	**52.1**
small lamp	56.8	**85.7**
spoon	42.7	**50.9**
vase	32.2	**55.9**
hammer	45.8	**45.8**
shark plushy	19.8	**25.1**

TABLE I: A selection of individual success rates out of the 35 objects we generate on in single-object scenes. Ours generation outperforms the baseline Get a Grip [14] due to improved grasp initializations from the VLM.

E. Diffusion Model Distillation

While the grasping pipeline can generate numerous candidate grasps from an object mesh, it is not directly deployable in real-world scenarios due to practical constraints. The generation process is quite slow, object mesh is not readily available, and the generation process often does not always return successful grasps. Inspired by [2], our diffusion model takes as input a Basis Point Set (BPS) which is a structured point cloud that can be readily obtained from the object mesh using a depth camera [52]. Additionally, the model receives a noisy grasp hypothesis. Through the denoising process, the diffusion model refines this noisy input into a feasible and executable grasp. This downstream diffusion model inherits both the geometric reasoning capabilities of the training pipeline and the semantic understanding provided by the vision-language model (VLM), as illustrated in Figure 4.

IV. EXPERIMENTAL SETUP

Datasets of grasps are generated on diverse objects in both single-object and clustered-scene settings, followed by extensive simulation to evaluate robustness. The evaluation addresses four key questions: (1) Can robust and stable grasps be produced on individual objects? (2) In clustered scenes, can the object of interest be identified and grasped without collision? (3) Do the resulting grasps exhibit natural, human-like qualities suitable for functional manipulation? (4) To what extent does semantic, part-level conditioning via segmentation improve grasp robustness and naturalness?

Task Setup. Two evaluation settings are considered. In the single-object case, 24 diverse objects from Get a Grip's dataset are used; each object is sampled at 5 scales, with 200 grasps generated by both our method and the baseline. For clustered scenes, 35 dense scenes from DexGraspNet2 are selected. Each scene contains on average 3–4 objects, with 3–4 segmentation prompts per object, and 200 grasps generated for each prompt–object pair. Baselines sample 256 grasps per scene. All objects are drawn from common daily manipulation tasks, and all grasps are executed using the LEAP Hand [**?**]. Finally, a diffusion model is trained to verify that the generated

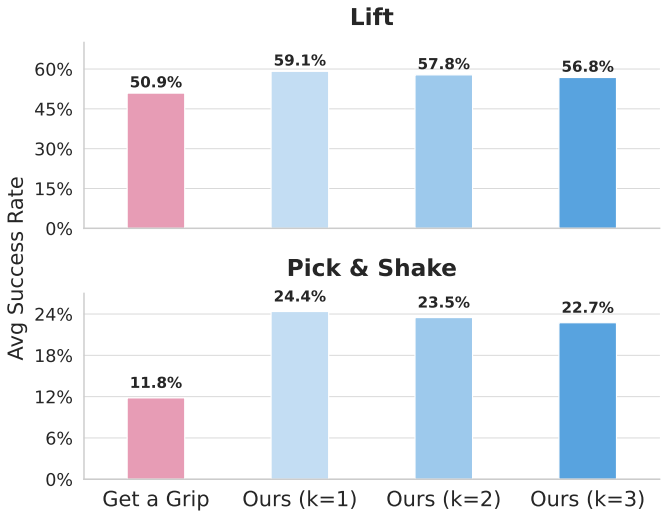

Fig. 5: Single Object evaluation in the Lift and Pick and Shake Task. Ours outperforms on the top three segmentation prompts compared to the Get a Grip baseline generation process due to the guidance that the prompt and the VLM provide on the grasping generation process.

grasps can be distilled into a policy operating directly on proprioceptive data obtainable in the real-world.

Simulation Evaluation. We test our grasps in IsaacGym [53]. For the single-object setting, two tasks are designed: Lift, which raises the wrist vertically to test grasp firmness, and Pick & Shake, which lifts the object slightly and applies perturbations to the wrist. A task is considered successful if the object's relative pose to the palm remains stable throughout execution. Collision checking is enforced during the entire process. For clustered scenes, we evaluate grasps only on the Lift task, since shaking in a dense environment often leads to trivial collisions.

V. RESULTS

A. Single Object Grasping

A useful grasp is not only robust but also natural, both of which can be achieved through our method. To demonstrate this, we evaluate our method against Get a Grip [14] on 35 diverse objects from their dataset used in daily scenarios. To assess robustness, grasps are evaluated in simulation under two tasks, Pick and Shake & Lift. As illustrated in Table II, IFG achieves higher success rates on both tasks, demonstrating that conditioning grasp generation on part-level segmentation produces more robust grasps. Table I further presents detailed success rates of a diverse set of objects from our data. Moreover, from qualitative comparisons, our grasps are more natural:

Method	Pick & Shake (%)	Lift (%)
Ours	16.14	51.11
Get a Grip	11.82	50.93

TABLE II: Single-object grasp generation evaluation in Isaac Gym. Our method outperforms Get a Grip by leveraging VLM-based part-level awareness. Successful grasps are filtered and used to train the diffusion model.

Method	Lift (%)
Ours	32.23
GraspTTA	25.64
ISAGrasp	32.51
DexGraspNet2	36.71

TABLE III: IFG can generate grasps with similar lift success rates to baseline models trained on preprocessed and filtered DexGraspNet2's Dataset, showing our strong grasp generation capabilities.

they are concentrated on the object regions that humans typically interact with in real-world use, while many of Get a Grip's grasps that pass simulation checks are not aligned with functional usage due to the absence of guidance during initialization. Shown in Figure 3, their grasps tend to grasp the head of a hammer since it covers a high percentage of the convex hull, while our grasps initialized on the segmented convex of the handle are functionally correct. With our method outperforming the baseline on both robustness and naturalness, we hypothesize that semantically conditioned grasps improve robustness because everyday objects are typically designed with affordances that support secure functional grasping, and semantic conditioning aligns grasp generation with these regions.

B. Multi-object Dense Scene Grasping

Daily scenarios are often not so simple as single object settings because they involve many clustered objects. A grasp proposal pipeline must therefore be able to identify the object of interest and generate firm grasps while avoiding collision with others. Get a Grip does not address multi-object scenes, so we compare against the crowded scene grasp generation models in DexGraspNet2[9]. DexGraspNet2 retargets GraspNet-1Billion data [12] into a diffusion model and adapts several single-object networks as baselines. Their approach ranks points on the scene point cloud with an MLP to propose grasp seeds, but cannot control which object is grasped. In addition, their ranking method tends to be biased toward easy targets, as shown in Figure 6. In contrast, our method selects via semantic segmentation prompts and avoids overfitting to easy-to-grasp regions. We evaluate IFG on clustered, dense scenes with harder objects from DexGraspNet2. Figure 1 shows our grasps on four scenes on the sides. Impressively, our synthetic generation method achieves a similar success rate compared to the baseline models distilled from preprocessed and filtered data, which is shown in Table III. A more detailed analysis done on individual objects across scenes is shown in Table IV.

The differences between ours and DexGraspNet2's reported performance is due to two reasons. (1) both methods lift objects by 20 cm, but DexGraspNet2 counts a grasp as successful if the object rises just 3 cm, even if it slips onto nearby objects. (2) More comprehensive testing: DexGraspNet2 reports only the top-confidence grasp per scene, usually on easy-to-grasp objects on the peripheral of the scene. We evaluate over 200 grasps per scene for their baselines. As shown in Figure 7, on harder, we outperforms them on occluded, harder-to-grasp objects that they grasp less frequently.

Object	DexGraspNet2	GraspTTA	ISAGrasp	Ours
Tomato Soup Can	47.8	38.3	**52.0**	45.5
Mug	33.2	26.9	22.6	**60.4**
Drill	32.1	20.8	36.4	**57.5**
Scissors	9.7	0.0	**33.7**	20.2
Screw Driver	0.0	8.3	**40.0**	22.0
Shampoo Bottle	50.6	25.4	18.8	**53.1**
Elephant Figure	23.6	29.6	24.2	**35.8**
Peach Can	**61.8**	28.0	55.3	60.3
Face Cream Tube	32.1	22.5	20.7	**35.5**
Tape Roll	22.7	13.9	9.8	**43.2**
Camel Toy	12.8	14.3	21.3	**21.8**
Body Wash	40.2	22.3	29.4	**58.3**

TABLE IV: Grasp success rates for crowded-scene evaluation on the lift task. The VLM enables IFG to focus on objects of interest and exceeds them in performance compared to baselines [9, 13, 4]

C. Grasp Generative Model

The method is modular, enabling plug-and-play replacement of both the segmentation and generation modules. For grasp generation, an attention-based conditional diffusion transformer (DiT) is trained to produce grasps conditioned on the object's BPS [52] representation computed from its point cloud, following an architecture similar to that used in Get a Grip. Grasps generated by this model, trained on semantically meaningful data, are compared against those produced by Get a Grip to highlight the benefits of semantic conditioning. (The model is trained on a single object at one scale, a bottle.)

VI. Conclusion and Limitations

We introduced IFG, a pipeline that combines the semantic understanding of vision-language models with the geometric precision of force closure grasping to generate functional and robust dexterous grasps. IFG harnesses internet-scale models to identify task-relevant object regions from visual information, uses them as semantic conditioning for energy based force closure optimization, and leverages simulation evaluation as a metric for robustness. As a result, IFG produces more natural and effective grasps than prior methods, particularly in cluttered environments. The resulting data is distilled into a diffusion model, enabling real-time grasp prediction from camera input without relying on hand-collected data.

Nonetheless, our work has limitations. First, our method does not account for dynamic objects since our segmentation is performed on images from a single timestep. Potential work can be done on extending our semantic segmentation pipeline for continuous video streaming. Additionally, our method is not suited for scenarios where non-force closure grasps are required. There is still much to be done in optimization based dexterous grasp generation.

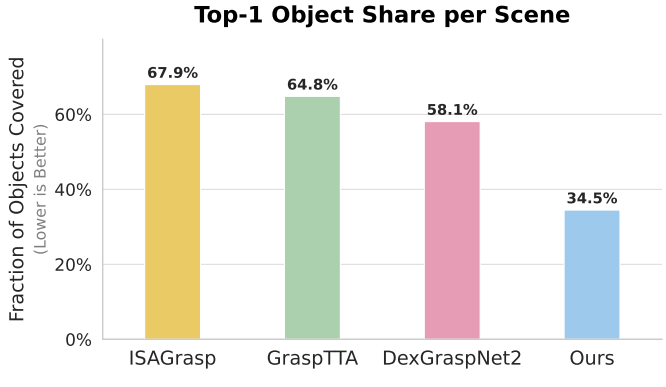

Fig. 6: When generating grasps, confidence-based methods generate most of their grasps on the easiest-to-grasp object. On the other hand, our method can be controlled to grasp any specific object due to segmentation conditioning. Therefore the easiest object is grasped less often.

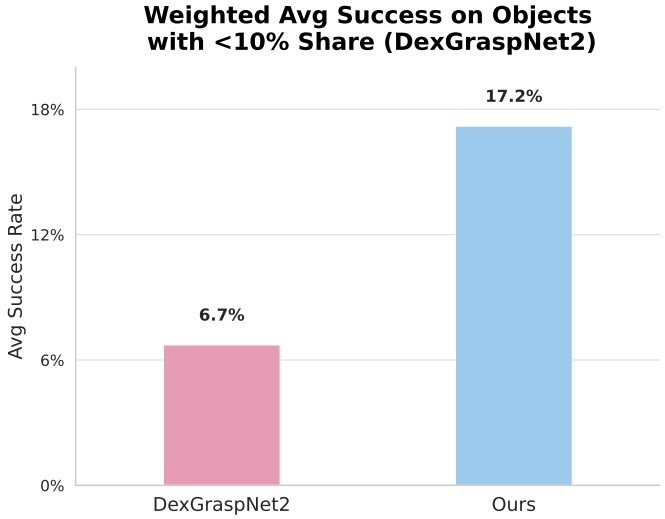

Fig. 7: DexGraspNet2's grasp generation model avoids hard-to-grasp objects. Our method concentrates more on these objects and achieves a higher success rate due to functional guidance from VLM-based segmentation.

References

[1] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4015–4026.

[2] Z. Weng, H. Lu, D. Kragic, and J. Lundell, "Dexdiffuser: Generating dexterous grasps with diffusion models," *IEEE Robotics and Automation Letters*, 2024.

[3] A. T. Miller and P. K. Allen, "Graspit! a versatile simulator for robotic grasping," *IEEE Robotics & Automation Magazine*, vol. 11, no. 4, pp. 110–122, 2004.

[4] Z. Q. Chen, K. Van Wyk, Y.-W. Chao, W. Yang, A. Mousavian, A. Gupta, and D. Fox, "Learning robust real-world dexterous grasping policies via implicit shape augmentation," *arXiv preprint arXiv:2210.13638*, 2022.

[5] D. Turpin, L. Wang, E. Heiden, Y.-C. Chen, M. Macklin, S. Tsogkas, S. Dickinson, and A. Garg, "Grasp'd: Differentiable contact-rich grasp synthesis for multi-fingered hands," in *European Conference on Computer Vision*. Springer, 2022, pp. 201–221.

[6] D. Turpin, T. Zhong, S. Zhang, G. Zhu, E. Heiden, M. Macklin, S. Tsogkas, S. Dickinson, and A. Garg, "Fast-grasp'd: Dexterous multi-finger grasp generation through differentiable simulation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 8082–8089.

[7] M. Attarian, M. A. Asif, J. Liu, R. Hari, A. Garg, I. Gilitschenski, and J. Tompson, "Geometry matching for multi-embodiment grasping," *arXiv*, 2023.

[8] R. Wang, J. Zhang, J. Chen, Y. Xu, P. Li, T. Liu, and H. 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)*. IEEE, 2023, pp. 11 359–11 366.

[9] J. Zhang, H. Liu, D. Li, X. Yu, H. Geng, Y. Ding, J. Chen, and H. Wang, "Dexgraspnet 2.0: Learning generative dexterous grasping in large-scale synthetic cluttered scenes," in *8th Annual Conference on Robot Learning*, 2024.

[10] D. P. Kingma, M. Welling *et al.*, "Auto-encoding variational bayes," 2013.

[11] P. Li, T. Liu, Y. Li, Y. Geng, Y. Zhu, Y. Yang, and S. Huang, "Gendexgrasp: Generalizable dexterous grasping," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 8068–8074.

[12] H.-S. Fang, C. Wang, M. Gou, and C. Lu, "Graspnet-1billion: A large-scale benchmark for general object grasping," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 444–11 453.

[13] H. Jiang, S. Liu, J. Wang, and X. Wang, "Hand-object contact consistency reasoning for human grasps generation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 11 107–11 116.

[14] T. G. W. Lum, A. H. Li, P. Culbertson, K. Srinivasan, A. D. Ames, M. Schwager, and J. Bohg, "Get a grip: Multi-finger grasp evaluation at scale enables robust sim-to-real transfer," *arXiv preprint arXiv:2410.23701*, 2024.

[15] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[16] C. R. Qi, H. Su, K. Mo, and L. 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*, 2017, pp. 652–660.

[17] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[18] C. Wang, H.-S. Fang, M. Gou, H. Fang, J. Gao, and C. Lu, "Graspness discovery in clutters for fast and accurate grasp detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 964–15 973.

[19] Y. Xu, W. Wan, J. Zhang, H. Liu, Z. Shan, H. Shen, R. Wang, H. Geng, Y. Weng, J. Chen *et al.*, "Unidexgrasp: Universal robotic dexterous grasping via learning diverse proposal generation and goal-conditioned policy," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4737–4746.

[20] T. Tao, M. K. Srirama, J. J. Liu, K. Shaw, and D. Pathak, "Dexwild: Dexterous human interactions for in-the-wild robot policies," *Robotics: Science and Systems (RSS)*, 2025.

[21] K. Shaw, Y. Li, J. Yang, M. K. Srirama, R. Liu, H. Xiong, R. Mendonca, and D. Pathak, "Bimanual dexterity for complex tasks," in *8th Annual Conference on Robot Learning*, 2024.

[22] Y.-W. Chao, W. Yang, Y. Xiang, P. Molchanov, A. Handa, J. Tremblay, Y. S. Narang, K. Van Wyk, U. Iqbal, S. Birchfield *et al.*, "Dexycb: A benchmark for capturing hand grasping of objects," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 9044–9053.

[23] P. Banerjee, S. Shkodrani, P. Moulon, S. Hampali, F. Zhang, J. Fountain, E. Miller, S. Basol, R. Newcombe, R. Wang *et al.*, "Introducing hot3d: An egocentric dataset for 3d hand and object tracking," *arXiv preprint arXiv:2406.09598*, 2024.

[24] Z. Fan, O. Taheri, D. Tzionas, M. Kocabas, M. Kaufmann, M. J. Black, and O. Hilliges, "Arctic: A dataset for dexterous bimanual hand-object manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 943–12 954.

[25] S. Brahmbhatt, C. Ham, C. C. Kemp, and J. Hays, "Contactdb: Analyzing and predicting grasp contact via thermal imaging," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 8709–8719.

[26] S. Brahmbhatt, C. Tang, C. D. Twigg, C. C. Kemp, and

J. Hays, "Contactpose: A dataset of grasps with object contact and hand pose," in *Computer Vision–ECCV 2020: 16th European Conference 2020, Proceedings*. Springer, 2020, pp. 361–378.

[27] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu *et al.*, "Ego4d: Around the world in 3,000 hours of egocentric video," in *IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 18 995–19 012.

[28] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price *et al.*, "The epic-kitchens dataset: Collection, challenges and baselines," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 4125–4141, 2020.

[29] Y. Rong, T. Shiratori, and H. Joo, "Frankmocap: Fast monocular 3d hand and body motion capture by regression and integration," *arXiv preprint arXiv:2008.08324*, 2020.

[30] Y. Ye, A. Gupta, and S. Tulsiani, "What's in your hands? 3d reconstruction of generic objects in hands," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 3895–3905.

[31] Y. Ye, P. Hebbar, A. Gupta, and S. Tulsiani, "Diffusion-guided reconstruction of everyday hand-object interaction clips," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 19 717–19 728.

[32] L. Shao, T. Migimatsu, Q. Zhang, K. Yang, and J. Bohg, "Concept2robot: Learning manipulation concepts from instructions and human demonstrations," *The International Journal of Robotics Research*, vol. 40, no. 12-14, 2021.

[33] A. S. Chen, S. Nair, and C. Finn, "Learning generalizable robotic reward functions from" in-the-wild" human videos," *arXiv preprint arXiv:2103.16817*, 2021.

[34] S. Bahl, A. Gupta, and D. Pathak, "Human-to-robot imitation in the wild," *RSS*, 2022.

[35] Y. J. Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, and A. Zhang, "Vip: Towards universal visual reward and representation via value-implicit pre-training," *arXiv preprint arXiv:2210.00030*, 2022.

[36] A. Sivakumar, K. Shaw, and D. Pathak, "Robotic telekinesis: Learning a robotic hand imitator by watching humans on youtube," *arXiv preprint arXiv:2202.10448*, 2022.

[37] A. Handa, K. Van Wyk, W. Yang, J. Liang, Y.-W. Chao, Q. Wan, S. Birchfield, N. Ratliff, and D. Fox, "Dexpilot: Vision-based teleoperation of dexterous robotic hand-arm system," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9164–9170.

[38] P. Sharma, D. Pathak, and A. Gupta, "Third-person visual imitation learning via decoupled hierarchical controller," *arXiv preprint arXiv:1911.09676*, 2019.

[39] K. Shaw, S. Bahl, and D. Pathak, "Videodex: Learning dexterity from internet videos," in *Conference on Robot Learning*. PMLR, 2023, pp. 654–665.

[40] A. Patel, A. Wang, I. Radosavovic, and J. Malik, "Learning to imitate object interactions from internet videos," *arXiv preprint arXiv:2211.13225*, 2022.

[41] L. Smith, N. Dhawan, M. Zhang, P. Abbeel, and S. Levine, "Avid: Learning multi-stage tasks via pixel-level translation of human videos," in *RSS*, 2020.

[42] K. Schmeckpeper, O. Rybkin, K. Daniilidis, S. Levine, and C. Finn, "Reinforcement learning with videos: Combining offline observations with interaction," *arXiv preprint arXiv:2011.06507*, 2020.

[43] S. Young, D. Gandhi, S. Tulsiani, A. Gupta, P. Abbeel, and L. Pinto, "Visual imitation made easy," *arXiv preprint arXiv:2008.04899*, 2020.

[44] J. Lee and M. S. Ryoo, "Learning robot activities from first-person human videos using convolutional future regression," in *CVPR Workshops*, 2017, pp. 1–2.

[45] A. Kannan, K. Shaw, S. Bahl, P. Mannam, and D. Pathak, "Deft: Dexterous fine-tuning for real-world hand policies," *arXiv preprint arXiv:2310.19797*, 2023.

[46] W. Huang, C. Wang, R. Zhang, Y. Li, J. Wu, and L. Fei-Fei, "Voxposer: Composable 3d value maps for robotic manipulation with language models," *arXiv preprint arXiv:2307.05973*, 2023.

[47] P. Sharma, B. Sundaralingam, V. Blukis, C. Paxton, T. Hermans, A. Torralba, J. Andreas, and D. Fox, "Correcting robot plans with natural language feedback," *arXiv preprint arXiv:2204.05186*, 2022.

[48] A. Agarwal, S. Uppal, K. Shaw, and D. Pathak, "Dexterous functional grasping," *arXiv preprint arXiv:2312.02975*, 2023.

[49] S. Nair, E. Mitchell, K. Chen, S. Savarese, C. Finn *et al.*, "Learning language-conditioned robot behavior from offline data and crowd-sourced annotation," in *Conference on Robot Learning*. PMLR, 2022, pp. 1303–1315.

[50] W. Yu, N. Gileadi, C. Fu, S. Kirmani, K.-H. Lee, M. G. Arenas, H.-T. L. Chiang, T. Erez, L. Hasenclever, J. Humplik *et al.*, "Language to rewards for robotic skill synthesis," *arXiv preprint arXiv:2306.08647*, 2023.

[51] J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. Florence, and A. Zeng, "Code as policies: Language model programs for embodied control," *arXiv preprint arXiv:2209.07753*, 2022.

[52] S. Prokudin, C. Lassner, and J. Romero, "Efficient learning on point clouds with basis point sets," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 4332–4341.

[53] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa *et al.*, "Isaac gym: High performance gpu-based physics simulation for robot learning," *arXiv preprint arXiv:2108.10470*, 2021.