Rethinking 6-DoF Grasp Detection: A Flexible Framework for High-Quality Grasping

Pengwei Xie[a,†], Siang Chen[a,†], Wei Tang[b,†], Kaiqin Yang[a], Guijin Wang[a,c,*]

[a]*Department of Electronic Engineering, Tsinghua University, Beijing 100084, PR China*
[b]*Department of Electronic Engineering, Shenzhen International Graduate School, Tsinghua University, Shenzhen 518071, PR China*
[c]*Shanghai AI Laboratory, Shanghai 200232, PR China*

Abstract

Robotic grasping is a fundamental skill for complex tasks and is essential to intelligent behavior. While most prior methods for general 6-DoF grasping focus on extracting scene-level semantic or geometric information, they often neglect adaptability to diverse downstream applications, such as target-oriented grasping. To address this, we rethink 6-DoF grasp detection from a grasp-centric perspective and propose **FlexLoG**, a versatile framework designed to unify scene-level and target-oriented grasping within a single system. Our framework integrates two core components: the *Flexible Guidance Module* and the *Local Grasp* model. The *Flexible Guidance Module* accommodates both global guidance (e.g., grasp heatmaps) and local guidance (e.g., visual-language grounding), ensuring robust grasp generation across varied task requirements. The *Local Grasp* model focuses on object-agnostic regional points, enabling precise, localized grasp predictions. Experiments demonstrate that **FlexLoG** surpasses existing methods, achieving a new state-of-the-art on the GraspNet-1Billion dataset. To validate the practical applicability, we reconstruct the dataset in a simulated digital twin environment, verifying the consistency of performance between simulated and benchmark results. Real-world robotic experiments across diverse scenarios further confirm the framework's efficacy, achieving a success rate exceeding 90%.

Keywords:

[†]Equal Contribution.
[*]Correspondence: wangguijin@tsinghua.edu.cn

Robotic grasp detection, Target-oriented grasping, Grasp representation, Grasp outcomes evaluation, Deep learning and computer vision

1. Introduction

Robotic grasping is fundamental for various complex robotic manipulation tasks in various fields, including manufacturing and service industries [1]. Despite its critical importance, the efficiency and quality of grasp detection in diverse scenarios remain unsatisfactory, particularly for target-oriented grasping.

Recent advances in deep learning have enabled data-driven methods for generating grasps of objects without the need for 3D models. Representative methods [2] generate planar grasps similar to the detection of rotated objects, achieving good performance in simple scenarios with high efficiency. However, this representation inherently constrains the gripper to be perpendicular to the camera plane, limiting its applicability in specific contexts.

Consequently, 6-DoF grasping has garnered extensive attention due to its broader applications. Pioneer methods employ either a sampling evaluation strategy [3, 4] or a direct regression paradigm [5] to predict grasp attributes, which are both time-consuming and inaccurate. Advanced works [6, 7, 8] excavate local features based on global information, implicitly divide grasp generation into two stages: ***where*** and ***how*** [7]. The ***where*** stage encodes global information to provide scene-level guidance, helping to locate grasps, and then the ***how*** stage generates refined grasps based on this information. Despite impressive performance, these approaches are flawed in two ways. Firstly, they are limited to integrating only with scene-level guidance in their ***where*** stage, without considering the suitability for different downstream applications, such as target-oriented grasping aiming to grasp specific objects or parts. For target-oriented grasping, [9, 10] first generate scene-level grasps and apply grasp filtering. These approaches introduce unnecessary computational load and risk interference from irrelevant objects, potentially leading to low-quality grasps or no grasp at all in the target area. Secondly, the quality of their generated grasps is still unsatisfactory, especially for unseen objects.

Unlike these scene-level methods, we rethink the 6-DoF grasp detection problem from a grasp-centric perspective and shift our attention from the scene level to the region level. We propose a novel flexible framework capable of handling both scene-level and target-oriented grasping. Concretely, our framework, named **FlexLoG**, is composed of a ***Flex***ible Guidance Module and a ***Lo***cal ***G***rasp (*LoG*) model. The *Flexible Guidance Module* is compatible with global and local guidance, aggregating multiple local regions. Subsequently, the *LoG* processes the multi-modal regional data and efficiently predicts grasps within each region.

Several key advantages arise from reformulating this problem from a local grasp-centric perspective. First, our framework can be guided by global or local guidance methods (e.g., scene-level heatmaps [8], object-level detection [11]), allowing for the efficient generation of high-quality scene-level or target-oriented grasps based on specific requirements. Second, the grasp-centric representation enables the network to learn grasp-related, object-agnostic geometric features, resulting in an over 23% improvement in performance for novel objects in the GraspNet-1Billion dataset [12]. Moreover, our approach demonstrates greater flexibility and superior generalization capabilities in novel scenarios compared to other scene-level methods, making it more adaptable and effective across a broader range of applications. Finally, we deploy our method on a real robot and conduct experiments in a variety of settings—***Cluttered***, ***Randomly Arranged***, ***Click-Grasp***, and ***Language-Grasp***—achieving impressive success rates of over 90%. These results highlight the applicability and effectiveness of our framework in real-world scenarios.

The major contributions of this paper are as follows.

(1) We develop a versatile and robust framework (FlexLoG) to unify scene-level and target-oriented grasping within a single system.

(2) We propose a *Flexible Guidance Module* that generates graspable regions for different grasp guidance modalities, including language instruction commands and interactive clicks.

(3) We design a target-oriented grasp network that integrates RGB, depth, and positional information through a multimodal de-differentiation module, and generates

local region grasps using a grasp predictor.

(4) We reconstruct 190 scenes from the GraspNet-1Billion dataset [12] in the Sapien simulator [13] to more accurately evaluate the grasping performance of the model.

(5) We implement **FlexLoG** in real-world robotic experiments, demonstrating its flexibility and applicability in various scenarios.

2. Related work

2.1. Scene-level grasping

Scene-level grasping means generating grasps for the entire scene in a target-agnostic manner. Some methods directly extract scene-level semantic or geometric information. [5] utilize PointNet++ [14] to encode scene points and then directly regress grasp attributes. Recent methods in grasp generation predominantly adopt a two-stage approach: ***where*** and ***how***.

Chen et al. [8] leverage ResNet [15] to generate heatmaps, guiding the identification of grasping locations in image space. Similarly, works like [12, 6, 7, 16, 17] utilize PointNet++ as an encoder to extract global per-point features, assigning each point a "graspness" or confidence score. Recognizing the importance of local information for effective grasping, these methods typically employ techniques like ball query or cylinder grouping to segment graspable regions around potential grasp centers. Chen et al. [8] further augment this process by integrating points within the graspable regions with corresponding semantic features. A vanilla PointNet [18] is then used to extract both geometric and semantic features. In contrast, other studies [6, 12, 7, 16, 17] employ Multilayer Perceptrons (MLP) to encode deeper features based on the global per-point features extracted by PointNet++. Based on these extracted features, various mechanisms are developed to predict different grasp attributes tailoring to the specific grasp representation.

In contrast to the aforementioned scene-level methods that generate scene-wide grasps, we reformulate the 6-DoF grasp detection problem in a grasp-centric view. **FlexLoG**, our innovative framework, stands as the first to predict high-quality grasps solely based on local information. This focus on local data significantly improves

the framework's ability to generalize to unseen objects, a benefit stemming from the object-agnostic characteristics of the regional points.

Besides, other methods such as Ten et al. [19], and Liang et al. [4] follow a sample-evaluation strategy and also sample grasps locally. Compared to our streamlined approach, their reliance on complex, hand-crafted features is more time-consuming and results in lower grasp quality.

2.2. Target-oriented grasping

Recently, several approaches [9, 10] focus on grasping specific objects in cluttered by integrating an additional segmentation branch. Such a strategy, though practical, often results in excess computational load and cannot consistently yield high-quality grasps for the intended target. [20] utilizes a visual grounding algorithm for object detection, using bounding box filters to obtain object-level point clouds. These clouds are then processed by a network [21] trained on scene-level data, which can result in suboptimal grasping due to a domain mismatch with the training data.

In recent years, large language models (LLMs) and vision language models (VLMs) have provided feasible grasp guidance for 6-DoF grasping of specific target objects. Liu et al. [22] generate scene-level object grasp candidates through Anygrasp [23] and utilize LangSAM [24] to extract grasp poses of specified objects. Qian et al. [25] use GPT-4o [26] to parse the potential grasping area of language instructions and segment the point cloud through LangSAM[24] or VLPart[27], subsequently generate grasps by Graspnet-baseline [12] and FGC-Graspnet [28]. Different from them, our framework, which is trained on regional grasp-centric data, can directly generate grasps on the target parts and seamlessly integrate various guidance methods, including object detection [29, 11] and instance segmentation [30, 31]. This approach allows for direct processing of local point clouds rather than the whole scene, eliminating redundant computations. The object-agnostic nature of our method ensures higher grasp quality on target objects. Besides, our framework can readily incorporate part affordance concepts [32] to facilitate grasping specific object parts.

In our prior work [33], we explored multimodal guidance for target object grasping. Here, we extend the method to enable scene-level grasping and enhance its validation

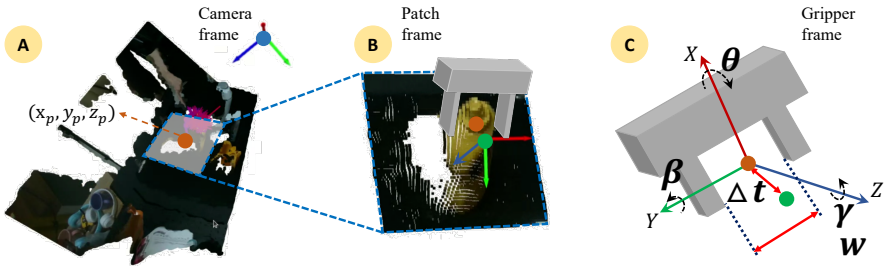

Figure 1: **Problem formulation of region-focal grasp detection.** A: The patch is cropped from the input RGB-D image. B: The local patches are transformed and normalized. C: The regional grasp representation as $(\Delta\mathbf{t}, \theta, \beta, \gamma, w)$.

through extensive experiments with datasets, simulations, and real-world test.

3. Problem formulation

Our task is to generate grasps in the corresponding scene according to different grasping guidance information. According to the different guidance areas, we can divide the grasping tasks into ***scene-level*** grasping and ***local-level*** grasping. Scene-level grasping requires global information guidance (such as heatmap) to generate grasps for all objects in the scene, and local-level grasping requires different instructions (such as language instructions) to specify local grasping targets and only generate grasps for them. Therefore, from the perspective of grasping objects, local-level grasping is equivalent to ***target-oriented*** grasping. It can be seen that the difficulty of our research task lies in accurately predicting grasps from scene information and grasping guidance information of different modalities.

Different from the previous scene-level grasp detection approaches [12, 8]. We decouple the task into two subtasks: extracting local graspable areas from different grasping guidance information and generating grasps from graspable areas. For grasping guidance instruction I, we first generate K local graspable RGBD patches; then we aim to predict 6-DoF grasps for each patch:

$$\boldsymbol{G_p} = \Phi(f_i | i = 1, ..., K), \tag{1}$$

where $f_i \in R^{N\times 3}$ is the i-th region patch which is cropped from the RGB-D image and centered at (x_p, y_p, z_p) in the camera frame, as shown in Figure 1(A). And $\Phi(\cdot)$ denotes *LoG* which predicts grasps $\boldsymbol{g_p} \in \boldsymbol{G_p}$ centered at (x_p, y_p, z_p):

$$\boldsymbol{g_p} = (\Delta\mathbf{t}, \theta, \beta, \gamma, w). \tag{2}$$

As shown in Figure 1(C), $(\theta, \beta, \gamma) \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ are grasp Euler angles in the gripper frame. θ represents the gripper in-plane rotation and β, γ represents the gripper orientation. To avoid collision in clutters, w denoting the grasp width is also predicted. Given that the guidance may be imprecise, to improve the robustness and accuracy, we introduce a 3D position offset, $\Delta\boldsymbol{t} = (\Delta x, \Delta y, \Delta z) \in [-2\text{ cm}, 2\text{ cm}]$, to finetune grasp position. Once $\boldsymbol{g_p}$ has been determined, the final grasp $\boldsymbol{g}$ on the target object can be represented as:

$$\boldsymbol{g} = (x_p + \Delta x, y_p + \Delta y, z_p + \Delta z, \theta, \beta, \gamma, w). \tag{3}$$

4. Method

4.1. Framework

The key motivation is to create a framework that is adaptable to various task-specific demands, from scene-level grasp detection to more precise target-oriented grasping. Unlike traditional methods that treat the entire image as a single region for grasp detection, our approach breaks down the problem into smaller, region-level tasks. This enables the system to focus on local regions of interest, improving both the efficiency and quality of grasp detection.

As shown in Figure 2, our flexible 6-DoF grasp detection framework, **FlexLoG**, consists of two key components: the ***Flex***ible Guidance Module (*FGM*) and the ***Lo***cal ***G***rasp (*LoG*) model. The flexibility of *FlexLoG* lies in its ability to handle both scene-level and local-level regions, as well as support multiple target-oriented guidance strategies, including object-level and part-level guidance.

The *FGM* leverages global or local guidance to sample potential grasp points from a monocular RGB-D image. Initially, a pre-trained grasp heatmap model [8] is used to generate a graspability heatmap, indicating regions with higher probabilities of successful grasps. Points from local peaks of the heatmap are then clustered to form distinct regions. Following the guidance provided by the *FGM*, the *LoG* model extracts

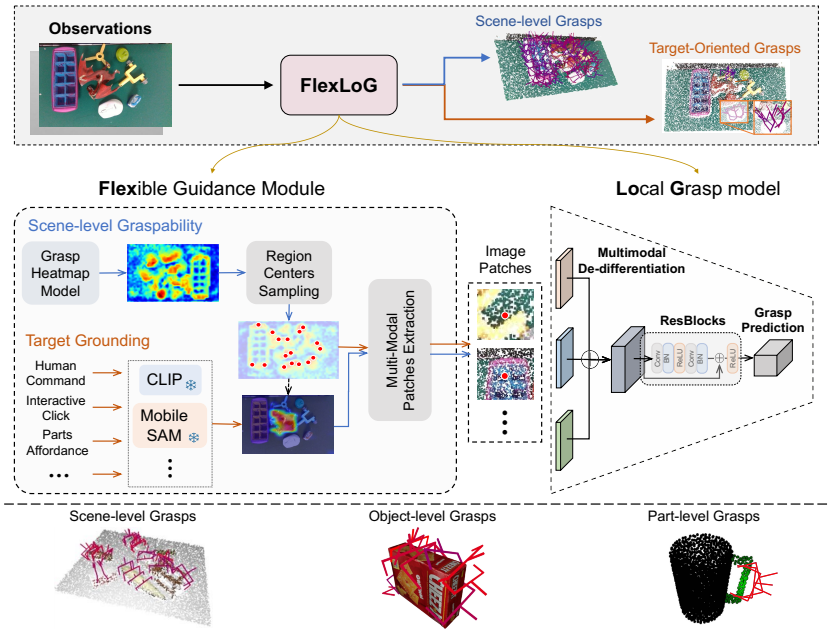

Figure 2: The architecture of **FlexLoG**. Taking a monocular observation RGB-D image as input, the *Flexible Guidance Module (FGM)* utilizes different guidance methods (e.g., grasp heatmap for scene-level guidance and human command like "I want the white mouse" for target grounding) to locate potential graspable regions and sample points from the grasp heatmap as regional centers. These centers are then clustered to obtain multiple multi-modal patches. The *Local Grasp (LoG)* model then extracts geometric features, predicts grasps on these patches, and aggregates them. Depending on the guidance method used in the FGM, the output is either scene-level or target-oriented grasps.

semantic and geometric features from these regions and predicts grasps aligned with their corresponding regional centers. Depending on the guidance method (scene-level, object-level, or part-level), the system refines the focus of the grasp prediction, either detecting all regions or targeting specific objects/parts.

The region-level, grasp-centric methodology of **FlexLoG** provides several advantages. By using smaller regions as the focus for grasp detection, the framework reduces the complexity of the task and increases the precision of grasp predictions. Additionally, the ability to integrate different types of guidance (e.g., object detection, part affordances, language instructions) within the *FGM* enhances the flexibility and adaptability of the system, making it effective in both general and specific tasks. Our approach also

significantly improves grasp quality, particularly in unseen scenarios, by leveraging a combination of regional grasping features and advanced guidance techniques.

In summary, **FlexLoG** offers a versatile, high-performance solution for 6-DoF grasp detection that adapts to diverse tasks and environments, improving grasp quality and generalization across both scene-level and target-oriented grasping scenarios.

4.2. Flexible guidance module

The *Flexible Guidance Module (FGM)* aims to identify and aggregate local regions with high graspability from the entire scene or specific targets. According to different downstream tasks, we categorize the guidance methods into two types: ***Scene-level Graspability*** and ***Target Grounding***.

As depicted in Figure 2, for scene-level grasping, we mainly adopt the grasp heatmap ϕ [8] to generate high-quality scene-level graspable regions. It indicate pixel-wise graspability and facilitate the sampling of points with high confidence, which are then used as regional centers for further aggregation. Specifically, given an RGB-D input image $I_{RGBD} \in \mathbb{R}^{H\times W\times 4}$, we get the graspability heatmap $M_g = \phi(I_{RGBD}) \in \mathbb{R}^{H\times W}$, with range from 0 to 1. Then, to avoid clustering the regional centers, we firstly divide M_g into $K_g \times K_g$ grids, each grid with the size of $\frac{H}{K_g} \times \frac{W}{K_g}$. And then we sample the highest pixel in each grid and get $K = K_g \times K_g$ regional centers accordingly.

Regarding target-oriented grasping, local guidance methods such as object detection [29] and semantic segmentation [30] can be utilized to sample regional centers from the predicted bounding boxes or segmentation masks. Furthermore, our **FlexLoG** framework is also compatible with other versatile guidance methods, such as part affordance [32], and even user-driven pixel selection through mouse clicks. In our work, we mainly introduce CLIP [34] and MobileSAM [35] to parse the human commands like "I want the toy dragon" from the RGB image input and get the target mask $M_t \in \mathbb{R}^{H\times W}$. Then, combining M_t and M_g, we get the target-level graspability heatmap and use the similar strategy to sample K regional centers. Note that the target can be both object like the toy dragon or object parts like the mug handle, which is flexible according to the demand.

Algorithm 1 Flexible region sampling strategy

Inputs:

$I_{RGBD} \in \mathbb{R}^{H\times W\times 4}$ - input RGB-D image

K_g - grid number along each side

$thres$ - region sampling threshold

Python-Style Pseudocode:

$M_g = \phi(I_{RGBD}) \in \mathbb{R}^{H\times W}$ # obtain the graspability heatmap

$M_t = \beta(I_{RGBD}) \in \mathbb{R}^{H\times W}$ # get target heatmap, β is CLIP/MobileSAM/...

$M_f = \text{dot}(M_g, M_t)$ # obtain the graspable target heatmap

$M'_f = \text{downsample}(M_f, K_g) \in \mathbb{R}^{K_g\times K_g}$ # downsample to get the grid heatmap

$C = \text{list}()$ # initialize empty region center list

for each grid: $G_{i,j} \in \mathbb{R}^{\frac{H}{K_g}\times\frac{W}{K_g}}, G_{i,j} \subseteq M_f, i = 1...K_g, j = 1...K_g$ **do**

  if $M'_f[i, j] > thres$ **then**

    C.append(argmax($G_{i,j}$)) # get the local maximas (centers) in each grid

  end if

end for

To achieve both scene-level and target-oriented grasping, our proposed Flexible Guidance Module (FGM) can seamlessly unify the two distinct types of guidance methods into a single flexible region sampling strategy. As is shown in Algorithm 1, this flexible region sampling strategy begins by obtaining the graspability heatmap and the target heatmap separately. The final graspable target heatmap is then derived through the dot product of these two heatmaps. Specifically, for scene-level grasping, we can easily set the target heatmap to the robot workspace to generate grasps across the entire scene. Subsequently, the grid-based sampling method previously mentioned for graspable heatmap sampling can be adapted to identify the centers of graspable regions on specific targets, without overly clustering the regional centers.

To obtain regional information in the graspable areas, we aggregate the regional information by extracting the multi-modal patches from all K regional centers. Specifically, we transform the depth image into XYZ map using the camera intrinsic matrix, denoted as $I_{xyz} \in \mathbb{R}^{H\times W\times 3}$. And we then fuse the XYZ map with the original RGB

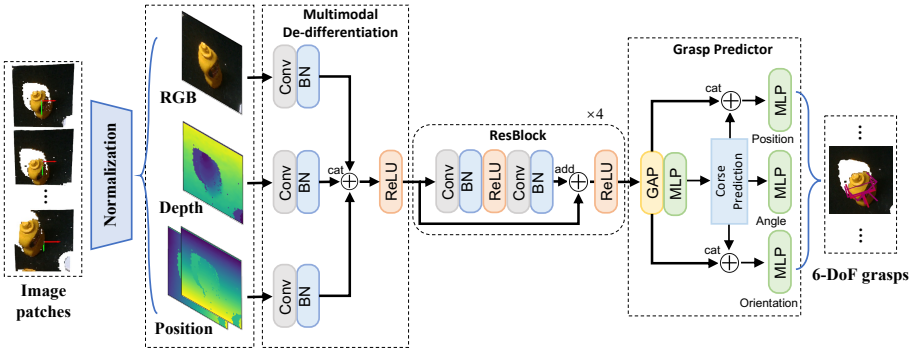

Figure 3: Detailed structure of proposed *LoG*.

image $I_{rgbxyz} \in \mathbb{R}^{H\times W\times 6}$. From the regional centers, we crop K multi-modal patches with patch size of $L \times L$, as $f_{rgbxyz} \in \mathbb{R}^{L\times L\times 6}$. RGB information gives the semantic features while XYZ map gives the geometric features. Notice that regional grasps are translation-invariant, i.e., the grasp detected in one region is not relevant to the region center's absolute translations. Thus, we normalize the patch-level XYZ map by translating the frame center to the patch center,

$$\bar{f}_{rgbxyz} = f_{rgb} \oplus (f_{xyz} - \text{Mean}(f_{xyz})). \tag{4}$$

4.3. Local grasp model

Traditional scene-level methods [6, 12, 7, 16, 8] process the scene-level inputs and then adopt filters to obtain target-oriented grasps. Although such a scene-level grasp detection pipeline is effective for general grasp detection, when faced with the task of target-oriented grasping, it introduces unnecessary computational costs for non-targeted parts and may struggle to generate feasible grasps on the target object in cluttered environments. Conversely, our method exclusively focuses on patch processing and region-focal grasp detection, considering only those grasps near the patch center. In this regard, we develop the ***Lo***cal ***G***rasp (***LoG***) model, a robust network designed for extracting local features in the target patches and predicting grasps. Our *LoG* is capable of predicting high-quality grasps solely utilizing local information. This advanced capability aligns seamlessly with the functionalities of our *FGM*. As illustrated

in Figure 3, our proposed *LoG* consists of three parts: Multimodal De-differentiation, ResBlock, and Grasp Predictor.

Multimodal De-differentiation. The quality of the depth image is significantly influenced by sensor noise, particularly at the edges, whereas RGB images are more resilient to noise and offer more precise object edges. Therefore, for robust and high-quality grasp detection, it is important to incorporate RGB information alongside the depth images. Moreover, since grasp poses are also related to object position, we include patch-level positional information, namely the X and Y coordinates of each pixel in the RGB-D images, calculated from the camera intrinsics. This enables the network to capture the positional relationship between grasps and objects.

However, we have noticed that features extracted from different modalities exhibit significantly different distributions. Taking inspiration from channel de-differentiation [36], we have investigated methods for aligning the modalities to process diverse inputs uniformly and ensure stable model training. In our approach, we implement multimodal alignment by applying a single-layer convolution and batch normalization before the feature extraction process:

$$f = \mathrm{ReLU}(\psi(f_{rgb}) \oplus \psi(\bar{f}_z) \oplus \psi(\bar{f}_{xy})), \tag{5}$$

where ψ is a one-layer convolution followed by batch normalization operation. $\oplus$ means feature concatenation. ReLU activation function is applied after multi-modal features are concatenated along the channel axis. By conducting modality normalization, we are able to process inputs with different modalities equally and train our network more stably. This minor improvement allows us to seamlessly incorporate RGB and positional data into our model, thus mitigating dimension mismatch and ensuring robust model training.

Grasp Predictor. After aligning the multi-modal inputs, we adopt a lightweight ResNet-18 as a backbone encoder to extract patch-level features efficiently. With the local features extracted, *LoG* employs a two-stage coarse-to-fine paradigm to predict region-focal grasps. Initially, the extracted features after Global Average Pooling (GAP) are processed by a Multilayer Perceptron (MLP) to generate grasp-related features and coarse grasp attributes. Then, in the second stage, the geometric features are

concatenated and then fed into three specialized MLP heads – ***Position Head***, ***Angle Head***, and ***Orientation Head*** to predict the final fine grasp attributes.

In the ***Position Head***, we approach position and width prediction as regression problems to prevent collisions with nearby objects and refine grasp placements within each region. In the ***Angle Head***, we predict the grasp in-plane rotation angle $\theta \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ by discretizing θ into 6 bins and conducting bin classification and residual regression. In the ***Orientation Head***, following the non-uniform anchor sampling strategy [8], our strategy views spatial rotation prediction as a multi-label classification task. For fair comparisons, we maintain the detailed settings as those in [8]. With 7 anchors each for (β, γ), we generate up to $N_A = 7^2 = 49$ possible grasps per region and preserve those with the highest scores.

Compared with scene-level methods [6, 12, 7, 16, 8], our LoG exclusively focuses on regional data and local geometric feature extraction, which significantly enhances grasp detection quality. A vital aspect of our strategy involves the adoption of regional point clouds, which frequently lack complete object shape information, as the network input. This compels the network to adapt to learning object-agnostic features, resulting in significantly enhanced generalization, especially in unseen scenarios.

4.4. Region-focal dataset generation

To effectively train *LoG*, we generate a new dataset of regional RGB-D patches with grasp labels derived from the GraspNet-1Billion [12] dataset. The dataset creation involves selecting potential grasp centers and cropping local regions around them. Intuitively, randomly sampling grasp center candidates across the entire space seems straightforward but is inefficient, resulting in a high proportion of invalid data. Due to domain shifts, sampling centers directly from the projected grasp labels also lead to suboptimal results during inference.

To address these challenges, we design a Gaussian-based strategy for center selection. As illustrated in Figure 4, the pipeline begins by projecting 6-DoF grasp annotations onto a 2D plane as 4-DoF planar grasps utilizing the camera intrinsics. We then encode each planar grasp center with the same Gaussian kernel to get the initial grasp location heatmap. Subsequently, to improve the robustness of our method, another

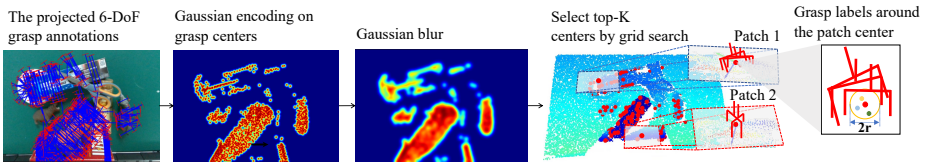

Figure 4: The pipeline of region-focal dataset generation. Grasp centers are sampled using the Gaussian-based strategy. Then, local neighboring points around each center are cropped as patches. Only the grasp labels within a radius of r from the patch center are preserved.

Gaussian filter is applied to the heatmap to generate a blurred heatmap representing the distribution of graspable areas, which introduces noise into the training process regarding the grasp center location. Similar to the sampling strategy in Algorithm 1, centers are selected from this heatmap via grid-based sampling and transformed into 3D points using the corresponding depth values.

Consistent with the grasp coverage criterion in [37, 9, 8], label processing involves preserving only those grasp labels within a $r = 2$ cm Euclidean distance from the selected region centers. This approach emphasizes *LoG*'s region-focal nature and ensures it focuses on generating grasps near regional centers, which is crucial for target-oriented grasping. To facilitate network training, points, and labels are transformed from the camera frame to the local region frame (we translate the origin to each patch center), normalizing the data distribution.

Through the above steps, we have constructed a substantial dataset comprising millions of region-focal and object-agnostic regional data, providing a robust foundation for training *LoG*.

4.5. Implementation details

As depicted in Section. 4.3, our *LoG* incorporates loss components of the prediction heads. Thus, the overall loss can be formulated with different loss terms as

$$\begin{aligned}
\mathbf{L} &= L_\theta^{cls} + L_\theta^{reg} + L_{\beta,\gamma} + L_w + L_{\text{offset}}, \\
L_\theta^{cls} &= \text{CrossEntropy}(\theta, \hat{\theta}), \\
L_\theta^{reg} &= \text{SmoothL1}(\Delta\theta, \Delta\hat{\theta}), \\
L_{\beta,\gamma} &= \text{CrossEntropy}([\beta, \gamma], [\hat{\beta}, \hat{\gamma}]), \\
L_w &= \text{SmoothL1}(w, \hat{w}), \\
L_{\text{offset}} &= \text{SmoothL1}(\text{offset}, \hat{\text{offset}}),
\end{aligned} \tag{6}$$

where (θ, $\Delta\theta$, β, γ, w, offset) represent the predicted grasp parameters and ($\hat{\theta}$, $\Delta\hat{\theta}$, $\hat{\beta}$, $\hat{\gamma}$, $\hat{w}$, $\hat{\text{offset}}$) represent the ground truth grasp parameters. L_θ^{cls} represents the cross-entropy loss for the gripper in-plane rotation angle θ classification. A smooth $L1$ loss L_θ^{reg} is adopted for the residual regression loss for θ prediction. $L_{\beta,\gamma}$ represents the multi-label classification loss calculated using focal loss [38] for (β, γ) classification. Two other smooth $L1$ losses L_w and L_{offset} are used to supervise the regression of the grasp width and center offset.

For *FGM*, we adopt the pretrained heatmap checkpoint [8] as our scene-level guidance since it is fully compatible with our *LoG*. For target grounding, we aggregated $K = 64$ local patches from the masked graspability heatmap. In addition, considering the trade-offs between inference time and detection performance, it is practicable to adjust the center number of local patches in different scenarios. In our experiments, we aggregate local patches with side length $L = 64$ pixels from the RGB-D images.

5. Experiment

5.1. Experimental design

To thoroughly evaluate the effectiveness of our proposed *FlexLoG*, we conduct extensive experiments in three stages: dataset evaluation, simulation experiments, and real robot trials. Through these studies, our aim is to address the following questions:

1) What is the quality of various grasp detection methods in the benchmark dataset? 2) Do our multi-modal inputs and key modules contribute effectively to grasp detection performance? 3) Do all methods perform consistently in the simulation environment, with results aligning with those from the dataset evaluation? 4) How well does our proposed method perform in real-world robotic experiments in different scenarios?

Concretely, we first evaluate the dataset in Section 5.2 to assess the performance of grasp detection, comparing it with various state-of-the-art methods for both scene-level and target-oriented grasp detection using the classical average precision metric. We then performed ablation studies to validate the contributions of key modules. Next, we design simulated scenarios based on the GraspNet-1Billion dataset setup, aiming to align the object grasping success rate with the dataset evaluation results and quantify performance. Finally, we performed real-world robot experiments to assess *FlexLoG*'s practical grasping capability. These systematic experiments and comparisons with other SOTA methods demonstrate our framework's flexibility and applicability.

5.2. Datasets experiments

Benchmark datasets. GraspNet-1Billion dataset [12] is widely used as a standard evaluation benchmark for 6-DoF grasp detection. It contains 97,280 RGB-D images captured in the real world from 190 cluttered scenes, along with over 1.1 billion grasp annotations. For a fair comparison, the data from the first 100 scenes are used for model training, and the data from the remaining scenes are used for testing. Specifically, test sets are further divided into three categories: 30 scenes with seen objects, 30 with unseen but similar objects, and 30 for novel objects. This setting can evaluate the generalizability of different methods. In our experiment, we utilize the method in Section. 4.4 to generate the region-focal data from the GraspNet-1Billion dataset. Around **6.5M** local patches are obtained from the training split to train the proposed *LoG*.

Evaluation metrics. Generally, for scene-level grasp prediction tasks, grasp detection or grasp generation methods usually predict multiple adequate grasp poses, and thus the percentage of true positive is the crucial component. Following the previous work [12], we adopt the **Average Precision (AP)** as our evaluation metric. As illustrated in [12], a grasp filter mechanism called grasp NMS (non-maximum suppression)

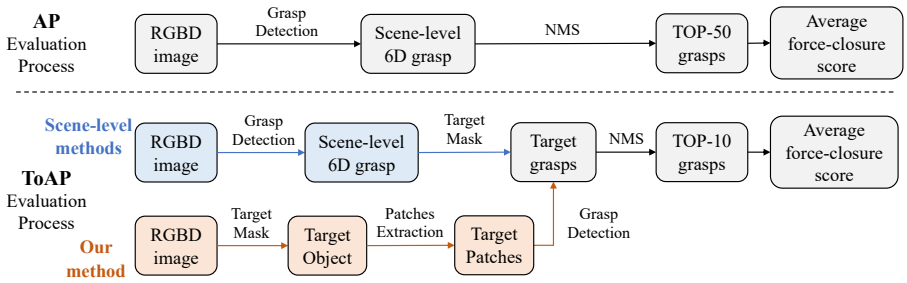

Figure 5: **AP** (Average Precision)[12] is adopted as the scene-level grasping evaluation metric. For target-oriented grasping, we design **TOAP** (Target-Oriented Average Precision) as the evaluation metric. **NMS** (non-maximum suppression) is the grasp post-processing process proposed by [12] to filter grasps.

is first adopted to filter dense grasp prediction results while maintaining the grasps with the highest scores in each local region. Then, the average Precision@50 by the force-closure scores [5] of the top 50 grasps is reported as the final AP score of the current scene.

When it turns to target-oriented grasping tasks, we modified the evaluation metric from **AP** to **T**arget-**O**riented **A**verage **P**recision (**TOAP**). For fair comparisons, we use the ground truth object segmentation mask from the GraspNet-1billion dataset to randomly select one target object, which is not fully occluded by others, as the target for each RGB-D image input. Since there are 512 RGB-D images with different cameras and viewpoints for each scene, and only approximately 10 objects per scene, all objects can be selected as targets. During the evaluation, we compute the distance from the detected grasp centers to the target object mesh model, retaining only the grasps near the target. Finally, due to reduced grasp diversity, we compute force-closure scores for the top 10 grasps after non-maximum suppression and use scores under different friction coefficients to calculate **TOAP**.

To summarize, the total evaluation process for grasping prediction tasks are shown in Figure 5. For more details about the grasp prediction evaluation, it can be referred to [12].

Experimental results. As illustrated in Table 1, *FlexLoG* makes significant gains and achieves the new state-of-the-art on the GraspNet-1billion benchmark. It is worth

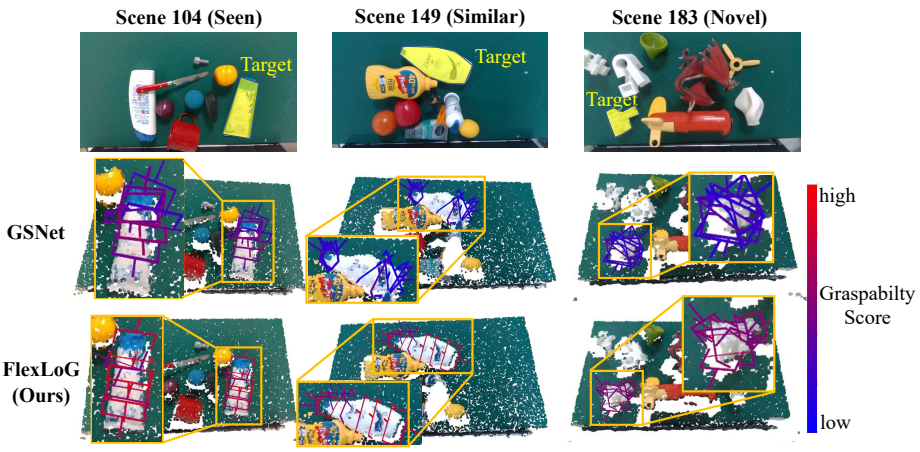

Figure 6: **Qualitative results covering seen/similar/novel test set.** Top 10 grasps on the target objects are displayed. Color implies the predicted graspability (red: high, blue: low).

noting that our grasp heatmap-guided approach achieves **10.4/9.83** and **5.73/3.89** performance gains on the similar and novel splits compared to the previous state-of-the-art method GSNet [7], which demonstrates that our region-focal and grasp-centric method has a more robust generalization to unseen scenarios.

For target-oriented grasping scenarios, We compare *FlexLoG*'s TOAP with former SOTAs, which are also trained on the GraspNet-1Billion dataset. Notably, Table 2 indicates that *FlexLoG* performs excellently on RGB-D images captured from RealSense and Kinect cameras, and outperforms current state-of-the-art by a large margin on all dataset splits, proving the efficacy of our proposed multimodal region-focal neural network.

We conduct grasp visualization of *FlexLoG* and GSNet [7] on three splits (seen objects, similar objects, and novel objects) of the test dataset. As depicted in Figure 6, *FlexLoG* exhibits superior grasp detection quality and achieves a higher grasp coverage rate on the target objects than the former SOTA GSNet. This enhancement provides the robot with more feasible grasp alternatives, thereby benefiting scene-level grasp planning and execution. Furthermore, it is noteworthy that *FlexLoG* enables the prediction of grasps that are better aligned with object centers and surfaces, avoiding possible collisions in cluttered scenes. The overall qualitative results underscore the validity of our

Table 1: Scene-level grasp performance on the GraspNet-1Billion dataset, reporting Average Precisions (APs) on the RealSense/Kinect split and method efficiency.

Method	**Seen ↑**	**Similar ↑**	**Novel ↑**	**Mean ↑**	**FPS ↑**
GPD [3]	22.87/24.38	21.33/23.18	8.24/9.58	17.48/19.05	-
GraspNet [12]	27.56/29.88	26.11/27.84	10.55/11.51	21.41/23.08	
RGB Matters [39]	27.98/32.08	27.23/30.40	12.25/13.08	22.49/25.19	2.3
REGNet [6]	37.00/37.76	27.73/28.69	10.35/10.86	25.03/25.77	2.2
TransGrasp [16]	39.81/35.97	29.32/29.71	13.83/11.41	27.65/25.70	-
GSNet [7]	67.12/63.50	54.81/49.18	24.31/19.78	48.75/44.15	~10
SBGrasp [40]	63.83/ -	58.46/ -	24.63/ -	48.97/ -	-
HGGD [8]	59.36/60.26	51.20/48.59	22.17/18.43	44.24/42.43	**28**
FlexLoG	**72.81/69.44**	**65.21/59.01**	**30.04/23.67**	**56.02/50.67**	26

framework for target-oriented grasping.

Ablation studies. Table 3 presents the ablation results conducted on the *FlexLoG*. We first conduct ablation of our region-focal grasp detection setting by increasing the distance threshold in Section 3 from 2 cm to 4 cm, which allows *FlexLoG* to generate grasps farther from the patch center. As the results show, keeping the region-focal grasp detection paradigm improves performance significantly. Constructing multimodal input by introducing XY coordinate maps (Positional Information) and RGB maps (Color Information) leads to considerable improvements. While geometric information from Z maps plays a crucial role in 6-DoF grasp detection, the additional positional information from XY maps and the inclusion of colors contribute to enhanced feature extraction. They are especially beneficial for generalizing unseen (similar and novel) scenes. Specifically, we compared the system's grasp detection performance with and without the inclusion of the proposed Multimodal De-differentiation module. The results

Table 2: Target-oriented grasp performance on the GraspNet-1Billion dataset, reporting Target-Oriented Average Precisions (TOAPs) on the RealSense/Kinect split.

Method	Seen	Similar	Novel	Mean
GraspNet [12]	22.64/13.28	20.63/12.67	8.35/3.85	17.21/9.93
SBGrasp [40]	38.04/ -	33.94/ -	15.97/ -	29.32/ -
HGGD [8]	38.91/34.82	34.70/28.77	16.73/12.02	30.11/25.20
GSNet [7]	44.80/34.81	37.67/29.61	18.06/12.82	33.51/25.75
FlexLoG	**51.84/49.60**	**46.62/40.03**	**23.74/19.58**	**40.73/36.40**

"-": Result unavailable

demonstrate that removing the de-differentiation module leads to a noticeable drop in performance, highlighting its role in enhancing grasp generalization and adaptability with multimodal information input.

5.3. Simulation experiments

While dataset testing offers an intuitive measure of the predictive power of different models, our primary objective is to evaluate how effectively the trained model performs in actual grasping trials. To this end, we construct various cluttered scenes in simulation and implement these methods in both scene clearance and target-object grasping experiments to rigorously validate their performance.

Experimental setup. To align the simulation results with our dataset evaluations, we replicate the 190 scene settings from the GraspNet-1Billion dataset [12] in the Sapien simulator [13], as illustrated in Figure 7. This setup offers additional advantages. First, this digital twin can expand the original dataset by enabling data collection from more camera viewpoints and parameter variations, thus increasing the dataset size with ease. Second, by adjusting table and object textures or creating more complex scene configurations and diverse object meshes, data diversity could be enhanced and further improves grasp detection performance.

Table 3: Ablation Experiments, showing TOAPs on RealSense Split.

RF	PI	CI	MD	Seen	Similar	Novel	Mean
				44.65	42.20	18.66	35.17
✓				50.05	43.39	21.86	38.43
✓	✓		✓	51.48	44.68	23.37	39.84
✓	✓	✓		51.03	46.23	22.75	40.00
✓	✓	✓	✓	**51.84**	**46.62**	**23.74**	**40.73**

RF: Region-Focal, **PI**: Positional Information, **CI**: Color Information, **MD**: Multimodal De-differentiation.

In the simulation environment, we configure a UR5e robotic arm equipped with a Robotiq 2F-85 gripper and a wrist-mounted RealSense RGB-D camera, replicating a standard robotic setup. To ensure smooth execution, we set the friction parameter of the physical simulation to 1. For scene clearance experiments, we generate scene-level grasps using various detection methods, selecting the grasp with the highest score for execution. The robot then plans and executes the grasp to remove the object from the clutter. The process iterates until all objects are picked or the robot experiences five consecutive failed grasp attempts (15 in our setting). We then calculate the following metrics: 1) **Grasp Success Rate (GSR)**[4], which measures the ratio of successful grasps to total attempts, and 2) **Scene Clearance Rate (SCR)**[4], representing the ratio of fully cleared scenes to total test scenes.

For a target-oriented grasping setting, in each test scene, we sequentially set each object as a target. We utilize the ground truth object mask as guidance to exclude the effects of guidance modules and concentrate solely on evaluating the performance of different grasp detectors. We also evaluate the Target-Oriented Grasp Success Rate (TOGSR) for different object splits.

Experimental results. We choose GSNet [7], HGGD [8], and AnyGrasp [23] as comparative baselines, noted for their exemplary performance. As indicated in Table

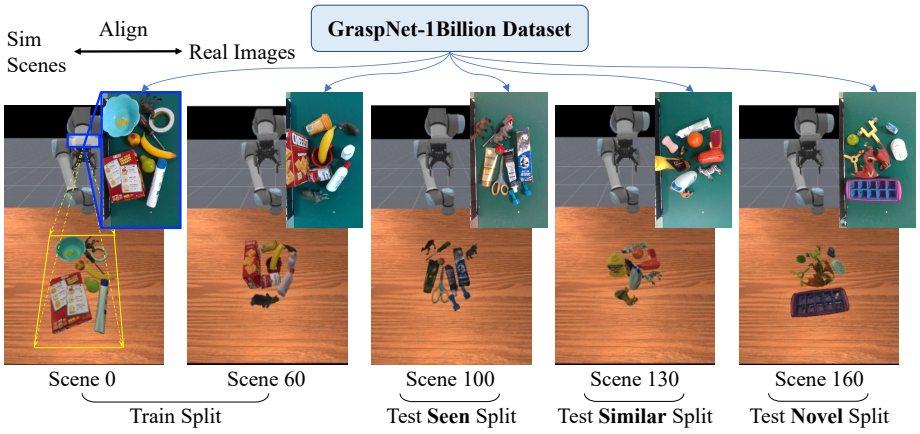

Figure 7: Simulation experimental settings. We present exemplar scenes from a total of 190 scenes. The simulation scenes are digital-twins of the scenes from the Graspnet-1billion Dataset [12], which are constructed to assess the actual grasp performance for different methods. And the results are aligned to the dataset evaluation results.

4, our *FlexLoG* achieves a grasp success rate of 66.2% and a scene clearance rate of 61.1%, which improves approximately 3.4% and 6.7% compared to the second-best baseline AnyGrasp [23]. In Table 4, our *FlexLoG* improves about 8.7% compared to the second-best baseline in the target-oriented setting. For the three baseline methods, we find that most failures occur because the filtered grasps around the target object are not of high quality (i.e., the high-quality grasps are clustered around the same object). In contrast, our model predicts grasps for the target object locally, significantly enhancing performance and efficiency.

For a more detailed visualization of our simulation grasping experiments, please refer to the first part of our video demo, available at `https://youtu.be/9g1K2-C8koc`.

5.4. Real robot experiments

Experimental setup. We also conduct real-world robot grasping experiments using a UR5e robot equipped with a Robotiq 2-finger parallel-jaw gripper. The Intel RealSense D435i camera is employed to acquire single-view RGB-D images. To ensure a fair comparison, the default settings of the camera, same as in former works [8, 12, 7], are utilized for capturing the raw RGB-D data. No additional filtering or

Table 4: Grasp Success Rate (GSR), Scene Clearance Rate (SCR) and Target-Oriented Grasp Success Rate (TOGSR) of simulation experiments for scene-level grasping and target-oriented grasping.

Scene	Metric	Method	Seen	Similar	Novel	Mean
Cluttered Scene	GSR	HGGD [8]	67.3%	57.9%	51.1%	58.8%
		GSNet [7]	63.4%	58.8%	58.6%	60.3%
		AnyGrasp [23]	66.5%	64.1%	57.8%	62.8%
		FlexLoG	**73.1%**	**65.0%**	**61.4%**	**66.5%**
	SCR	HGGD [8]	56.7%	46.7%	40.0%	47.8%
		GSNet [7]	60.0%	40.0%	46.7%	48.9%
		AnyGrasp [23]	60.0%	50.0%	53.3%	54.4%
		FlexLoG	**66.7%**	**56.7%**	**60.0%**	**61.1%**
Target-oriented	TOGSR	HGGD [8]	53.2%	58.9%	58.0%	56.7%
		GSNet [7]	66.4%	61.6%	60.4%	62.8%
		AnyGrasp [23]	62.1%	62.8%	61.8%	62.2%
		FlexLoG	**71.1%**	**71.7%**	**69.8%**	**70.9%**

post-processing techniques are applied to enhance the quality of the data.

As illustrated in Figure 8, we evaluate the performance of *FlexLoG* and AnyGrasp [23] in both *scene-level* and *target-oriented* grasping settings using different scene clearance and target-oriented grasping experiments. For the former, we create two realistic scenarios: *Cluttered* and *Random Arranged*. For the latter, we use *Click-Grasp* and *Language-Grasp* as guidance for target object selection.

In the *Cluttered* setting, following [4], we shake a box containing all objects, ensuring that the poses of the objects are randomized. In the *Random Arranged* setting, we simulate common object arrangements found in daily life, such as a bottle standing upright, before randomly arranging these objects on a tabletop. In the *Click-Grasp* set-

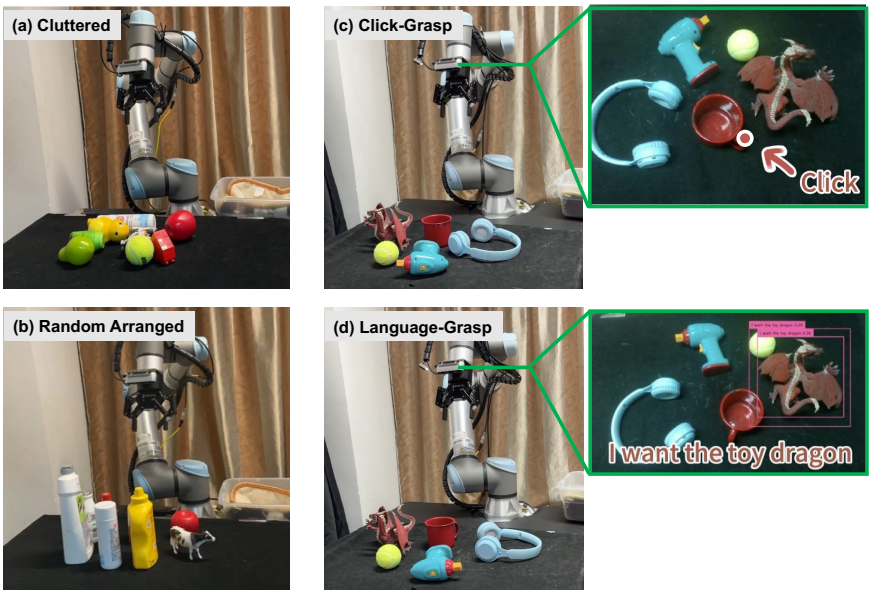

Figure 8: The settings of real-world robot experiments. (a) **Cluttered**: The poses of objects are randomized by shaking them in a box. (b) **Random Arranged**: Objects are arranged randomly to simulate typical configurations found in daily life. (c) **Click-Grasp**: Points selected by clicking are used as guidance for target-oriented grasping. (d) **Language-Grasp**: Language commands are employed to specify the target object.

ting, a random object is selected, and a click on a specific part generates grasps based on the points in the local region. In the *Language-Grasp* setting, natural language commands are used to specify the target object. In both settings, the target object mask is used to filter scene-level grasps generated by the AnyGrasp model, with the grasp having the highest score selected for its evaluation.

We evaluate the performance of scene clearance tasks using the **GSR** and **SCR**, consistent with the simulation metrics. For the target-oriented setting, success is defined by the successful grasp of the target object.

Experimental results. It is worth noting that, in the *Cluttered* and *Random Arranged* settings, Table 5 shows that our method achieves an average grasp success rate of 95% across all 12 test scenarios, with a 100% scene completion rate. This demonstrates that *FlexLoG* can effectively generalize to the real world and efficiently generate

Table 5: Results of real scene clearance experiments.

Scene	FlexLoG		AnyGrasp[23]	
	GSR	**SCR**	**GSR**	**SCR**
Cluttered	44 / 46 = 96%	6 / 6 = 100%	44 / 51 = 86%	5 / 6 = 83%
Random Arranged	48 / 51 = 94%	6 / 6 = 100%	48 / 55 = 87%	6 / 6 = 100%
Total	**95%**	**100%**	87%	92%

high-quality grasps. Failures are observed in scenarios where objects have smooth surfaces (e.g. bottles), causing the gripper to slip over the object, or when the gripper collides with other objects in the clutter.

In the *Click-Grasp* and *Language-Grasp* settings, Table 6 indicates that our method achieves an average grasp success rate of 91% across all target-oriented scenarios. This suggests that our model is adaptable to more flexible, target-oriented grasping tasks in real-world environments. In the current experiments, the primary cause of failure is unstable contact between the gripper and the target object, while additional failures arise from incorrect identification by the GroundingDINO model [29].

Compared with AnyGrasp [23], our FlexLoG outperforms in both scene-level and target-oriented grasping tasks, highlighting its flexibility and effectiveness in real-world applications.

For a more detailed visualization of our real-world grasping experiments, please refer to the second part of our video demo, available at `https://youtu.be/9g1K2-C8koc`.

5.5. Limitation and Discussion

Through both simulation and real-world experiments, we have been able to categorize the failure cases of our method and delve deeper into understanding its limitations. Our analysis reveals that the primary failure cases are linked to imprecise target localization and the detection of unfeasible grasp poses, both largely attributed to sensor noise. Our system relies on RGB-D image input, and its performance hinges on the quality of these images. In situations characterized by significant noise—such

Table 6: Results (grasp success rate) of real target-oriented grasping experiments.

Scene	FlexLoG	AnyGrasp[23]
Click-Grasp	48 / 51 = 94%	48 / 58 = 83%
Language-Grasp	39 / 45 = 87%	39 / 50 = 78%
Total	**91%**	81%

as motion blur in RGB images, which adversely affects graspable heatmap generation and target object localization, or gaps in depth images for transparent or reflective objects, which hinder local grasp generation and collision detection—our method's performance may suffer. To address the challenges posed by sensor noise, we propose several potential solutions, including the use of depth completion or denoising networks as preprocessing tools to enhance the quality of depth data. Additionally, we recommend fusing multi-view observations through 3D reconstruction techniques to improve spatial consistency and incorporating uncertainty-aware grasp prediction to downweight low-confidence input regions.

A notable limitation inherent in our current research pertains to the dataset, which has been developed specifically for two-finger grippers of a predetermined size. Consequently, the performance of our model may demonstrate suboptimal results when applied to two-finger grippers of different sizes, as well as a lack of applicability to alternative gripper configurations, such as three-finger grippers or dexterous hands. In light of the advancements in dexterous robotic manipulators and the emergence of more generalized object manipulation methodologies, it is imperative to explore strategies to extend our existing framework for broader applicability, enabling deployment across a variety of robotic platforms. To this end, we propose several prospective directions for future research. These include the development of gripper-agnostic representations within local regions, the utilization of domain randomization to enhance the transferability of our approach across diverse gripper types, and the establishment of simulation benchmarks that integrate a broader array of digital twins representing various

manipulation systems.

6. Conclusion

In this paper, we rethink the 6-DoF grasp detection problem and propose a flexible 6-DoF grasp framework, **FlexLoG**. Through a novel grasp-centric approach, the designed *Local Grasp* model can be integrated with global or local guidance methods for scene-level and target-oriented grasping tasks. Our framework achieves state-of-the-art performance on the GraspNet-1Billion dataset. To thoroughly evaluate our robot grasping system, we design tests across datasets, simulations, and real-world experiments. A comparison with other recent state-of-the-art methods demonstrates that our system can efficiently perform object grasping tasks. The quantitative results from real-world robot grasping experiments demonstrate the flexibility of our method, showcasing its potential for various future applications. In future work, we intend to expand our system's capabilities to tackle more challenging objects, such as transparent, reflective, and deformable items, as well as to accommodate more complex end-effectors, including suction cups and dexterous hands.

References

[1] I. Gonzalez-Diaz, J. Benois-Pineau, J.-P. Domenger, D. Cattaert, A. de Rugy, Perceptually-guided deep neural networks for ego-action prediction: Object grasping, Pattern Recognition 88 (2019) 223–235.

[2] S. Kumra, S. Joshi, F. Sahin, Antipodal robotic grasping using generative residual convolutional neural network, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2020, pp. 9626–9633.

[3] A. ten Pas, M. Gualtieri, K. Saenko, R. Platt, Grasp pose detection in point clouds, The International Journal of Robotics Research 36 (13-14) (2017) 1455–1473.

[4] H. Liang, X. Ma, S. Li, M. Görner, S. Tang, B. Fang, F. Sun, J. Zhang, Pointnetgpd: Detecting grasp configurations from point sets, in: 2019 International Conference on Robotics and Automation (ICRA), IEEE, 2019, pp. 3629–3635.

[5] Y. Qin, R. Chen, H. Zhu, M. Song, J. Xu, H. Su, S4g: Amodal single-view single-shot se (3) grasp detection in cluttered scenes, in: Conference on robot learning, PMLR, 2020, pp. 53–65.

[6] B. Zhao, H. Zhang, X. Lan, H. Wang, Z. Tian, N. Zheng, Regnet: Region-based grasp network for end-to-end grasp detection in point clouds, in: 2021 IEEE international conference on robotics and automation (ICRA), IEEE, 2021, pp. 13474–13480.

[7] C. Wang, H.-S. Fang, M. Gou, H. Fang, J. Gao, 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. 15964–15973.

[8] S. Chen, W. Tang, P. Xie, W. Yang, G. Wang, Efficient heatmap-guided 6-dof grasp detection in cluttered scenes, IEEE Robotics and Automation Letters 8 (8) (2023) 4895–4902. doi:10.1109/LRA.2023.3290513.

[9] M. Sundermeyer, A. Mousavian, R. Triebel, D. Fox, Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes, in: 2021 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2021, pp. 13438–13444.

[10] Z. Liu, Z. Wang, S. Huang, J. Zhou, J. Lu, Ge-grasp: Efficient target-oriented grasping in dense clutter, in: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2022, pp. 1388–1395.

[11] C.-Y. Wang, A. Bochkovskiy, H.-Y. M. Liao, Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 7464–7475.

[12] H.-S. Fang, C. Wang, M. Gou, 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. 11444–11453.

[13] F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang, et al., Sapien: A simulated part-based interactive environment, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 11097–11107.

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

[15] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[16] Z. Liu, Z. Chen, S. Xie, W.-S. Zheng, Transgrasp: A multi-scale hierarchical point transformer for 7-dof grasp detection, in: 2022 International Conference on Robotics and Automation (ICRA), IEEE, 2022, pp. 1533–1539.

[17] X. Liu, Y. Zhang, H. Cao, D. Shan, J. Zhao, Joint segmentation and grasp pose detection with multi-modal feature fusion network, in: 2023 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2023, pp. 1751–1756.

[18] C. R. Qi, H. Su, K. Mo, 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.

[19] A. Ten Pas, R. Platt, Using geometry to detect grasp poses in 3d point clouds, Robotics Research: Volume 1 (2018) 307–324.

[20] Y. Lu, Y. Fan, B. Deng, F. Liu, Y. Li, S. Wang, Vl-grasp: a 6-dof interactive grasp policy for language-oriented objects in cluttered indoor scenes, in: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2023, pp. 976–983.

[21] Y. Lu, B. Deng, Z. Wang, P. Zhi, Y. Li, S. Wang, Hybrid physical metric for 6-dof grasp pose detection, in: 2022 International Conference on Robotics and Automation (ICRA), IEEE, 2022, pp. 8238–8244.

[22] P. Liu, Y. Orru, C. Paxton, N. M. M. Shafiullah, L. Pinto, Ok-robot: What really matters in integrating open-knowledge models for robotics, arXiv preprint arXiv:2401.12202 (2024).

[23] H.-S. Fang, C. Wang, H. Fang, M. Gou, J. Liu, H. Yan, W. Liu, Y. Xie, C. Lu, Anygrasp: Robust and efficient grasp perception in spatial and temporal domains, IEEE Transactions on Robotics (2023).

[24] L. Medeiros, Language segment-anything, `https://github.com/luca-medeiros/lang-segment-anything` (2023).

[25] Y. Qian, X. Zhu, O. Biza, S. Jiang, L. Zhao, H. Huang, Y. Qi, R. Platt, Thinkgrasp: A vision-language system for strategic part grasping in clutter, in: 8th Annual Conference on Robot Learning, 2024.

[26] OpenAI, Gpt-4o, accessed: 2024-10-27 (2023).
URL `https://openai.com`

[27] P. Sun, S. Chen, C. Zhu, F. Xiao, P. Luo, S. Xie, Z. Yan, Going denser with open-vocabulary part segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 15453–15465.

[28] Y. Lu, B. Deng, Z. Wang, P. Zhi, Y. Li, S. Wang, Hybrid physical metric for 6-dof grasp pose detection, in: 2022 International Conference on Robotics and Automation (ICRA), 2022, pp. 8238–8244. doi:10.1109/ICRA46639.2022.9811961.

[29] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, Q. Jiang, C. Li, J. Yang, H. Su, et al., Grounding dino: Marrying dino with grounded pre-training for open-set object detection, in: European Conference on Computer Vision, Springer, 2024, pp. 38–55.

[30] 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.

[31] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18, Springer, 2015, pp. 234–241.

[32] S. Deng, X. Xu, C. Wu, K. Chen, K. Jia, 3d affordancenet: A benchmark for visual object affordance understanding, in: proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 1778–1787.

[33] P. Xie, S. Chen, Y. Dai, D. Hu, K. Yang, G. Wang, Target-oriented object grasping via multimodal human guidance, in: European Conference on Computer Vision, Springer, 2025, pp. 305–322.

[34] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in: International conference on machine learning, PMLR, 2021, pp. 8748–8763.

[35] C. Zhang, D. Han, Y. Qiao, J. U. Kim, S.-H. Bae, S. Lee, C. S. Hong, Faster segment anything: Towards lightweight sam for mobile applications, arXiv preprint arXiv:2306.14289 (2023).

[36] Z. Yang, Y. Sun, S. Liu, X. Qi, J. Jia, Cn: Channel normalization for point cloud recognition, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16, Springer, 2020, pp. 600–616.

[37] A. Mousavian, C. Eppner, D. Fox, 6-dof graspnet: Variational grasp generation for object manipulation, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 2901–2910.

[38] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2980–2988.

[39] M. Gou, H.-S. Fang, Z. Zhu, S. Xu, C. Wang, C. Lu, Rgb matters: Learning 7-dof grasp poses on monocular rgbd images, in: 2021 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2021, pp. 13459–13466.

[40] H. Ma, D. Huang, Towards scale balanced 6-dof grasp detection in cluttered scenes, in: Conference on Robot Learning, PMLR, 2023, pp. 2004–2013.