

# RealVLG-R1: A Large-Scale Real-World Visual-Language Grounding Benchmark for Robotic Perception and Manipulation

## Supplementary Material

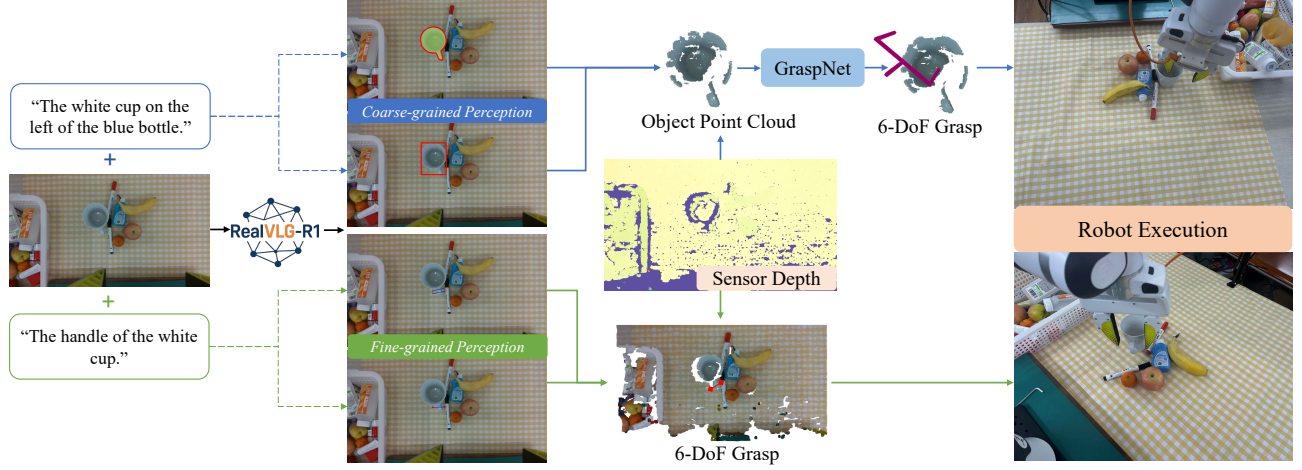


Figure 1. **Overview of the RealVLG-R1 deployment for real-world Visual-Language Grasping tasks.** RealVLG-R1 produces multi-granularity visual–language outputs, which can be leveraged in two complementary grasping strategies: (a) coarse-grained, object-centric grasping, where segmentation masks or bounding boxes are projected into 3D point clouds to generate 6-DoF grasp poses via a 3D grasping module; (b) fine-grained, part-level grasping, where 2D grasp predictions are directly transformed into executable 6-DoF poses using depth and camera parameters, enabling semantically precise manipulation. This design supports hierarchical control from global geometry to detailed semantic structures.

### 1. Real-world Visual-Language Grasping

As illustrated in Fig. 1, this section describes how the multi-granularity visual–language understanding capabilities of RealVLG-R1 are deployed in real-world Visual-Language Grasping tasks. To address different robotic application requirements, we design two complementary grasping strategies: a coarse-grained, object-centric approach and a fine-grained, part-level approach, enabling hierarchical control from global geometry to detailed semantic structures.

**Coarse-Grained Grasping (Object-Level).** In the coarse-grained strategy, RealVLG-R1 outputs object bounding boxes (Bbox) or segmentation masks (Seg.), which are used to generate a binary mask  $M$ . The corresponding depth region is first extracted from the sensor depth  $D$ :

$$D_{\text{obj}} = D \otimes M, \quad (1)$$

and then projected into 3D space using the camera intrinsic and extrinsic parameters :

$$\hat{P}_{\text{obj}} = \pi^{-1}(D_{\text{obj}}, K, T), \quad (2)$$

where  $\pi^{-1}(\cdot)$  denotes the standard depth-to-point-cloud back-projection operation. The resulting object point cloud  $\hat{P}_{\text{obj}}$  is subsequently input to a 3D grasping model such as GraspNet [2] to generate multiple candidate 6-DoF grasp poses. A grasp planner then selects the optimal pose, which is executed by the robotic manipulator. This pipeline establishes a staged process of “2D visual–language localization  $\rightarrow$  point cloud reconstruction  $\rightarrow$  6-DoF grasp generation,” demonstrating stability in conventional object grasping scenarios. However, grasp quality is constrained by depth noise and occlusions, and the generated grasp poses typically lack fine-grained control over specific semantic parts.

**Fine-Grained Grasping (Part-Level).** In the fine-grained strategy, RealVLG-R1 directly predicts 2D grasp priors on the image plane, including rectangular grasp poses or grasp contact points. To convert these 2D predictions into executable 6-DoF poses, each pixel  $(u, v)$  is first projected into the camera coordinate system using the corresponding sensor depth value  $d$  and camera intrinsics  $K$ :

$$\mathbf{p}_c = d \cdot K^{-1} \begin{bmatrix} u & v & 1 \end{bmatrix}^T \quad (3)$$

For a predicted grasp rectangle  $\mathcal{B} = \{(u_i, v_i)\}_{i=1}^4$ , the four corner points are projected to obtain  $\{\mathbf{p}_c^{(i)}\}_{i=1}^4$ , from which the grasp direction and normal vector are computed to define a local grasp coordinate frame  $R_c$  and center position  $\mathbf{t}_c$ . Finally, the local grasp pose is transformed into the robot coordinate system using the camera-to-robot extrinsics  $[R_{cam}^{rob} | \mathbf{t}_{cam}^{rob}]$ :

$$\mathbf{t}_{rob} = R_{cam}^{rob} \mathbf{t}_c + \mathbf{t}_{cam}^{rob}, \quad R_{rob} = R_{cam}^{rob} R_c. \quad (4)$$

This procedure establishes a direct mapping from 2D model predictions to executable 6-DoF grasp poses, enabling semantically precise part-level grasping, such as “grasping the handle of a cup”, while maintaining both grasping efficiency and semantic consistency.

Overall, the two strategies offer complementary advantages: coarse-grained grasping provides a stable and generalizable approach with clear staged execution but limited semantic specificity, whereas fine-grained grasping enables semantically aligned, part-level manipulation but imposes higher requirements on geometric accuracy and depth estimation. In future work, we aim to extend RealVLG-R1 to full 3D Visual-Language Grounding and Grasping, allowing the system to generate geometrically accurate and semantically consistent executable grasp poses directly in 3D space, thereby enabling more robust and precise language-driven robotic manipulation.

## 2. Details of RealVLG

### 2.1. Details of RealVLG-11B Dataset

In Fig. 2 of the main text, the Object Meta Description and Localization Description of each object are defined by Prompt 2.1 and Prompt 2.2, respectively, and are fed into GPT-4o [4] to generate diverse yet semantically consistent linguistic descriptions. The Detection Prompt (Prompt 2.3) is subsequently input to Qwen-VL-Max [1] to predict the bounding boxes of the target objects, thereby achieving structured alignment between linguistic and visual modalities. Furthermore, as illustrated in Fig. 2, the human verification phase involves annotators reviewing and refining the automatically generated results via the Human-Verification System, ensuring both accuracy and consistency of the annotations. The entire annotation process was collaboratively conducted by three annotators over a period of approximately five months, ultimately yielding a high-quality, multi-granularity vision-language annotation dataset.

#### Prompt 2.1: Object Metadata Description Prompt

Look at the provided images of the object "obj\_name" and write a concise English description, including its color, shape, and category. Begin the sentence with "A" or "An", avoid using any verbs (e.g., do not use “is”, “has”, etc.), and keep the description no longer than 10 words.

#### Prompt 2.2: Location Description Prompt

Analyze the following image and provide a concise description for each object listed in the object annotations. Object Annotations (use as reference only, focus on visible objects): {obj\_ann\_str}.

Guidelines:

1. Treat the object annotations only as reference. If the object is not clearly visible in the image, write "Not visible".
2. Focus on the object’s color, shape, size, and spatial relationships with other visible objects.
3. Do not invent or describe objects that are not present in the image.
4. Return your output strictly as a JSON array. Each element must follow this format: {"ObjectID": "<id>", "Description": "<short description>"}

#### Prompt 2.3: Detection Prompt

You are given an image and a list of object descriptions. For each object, locate it in the image and return the result as a JSON array with the following structure:

```
[
  {
    "object_id":
      "<original_object_id>",
    "bbox_2d":
      [x_min, y_min, x_max, y_max],
    "label":
      "<2-5 words>"
  },
  ...
]
```

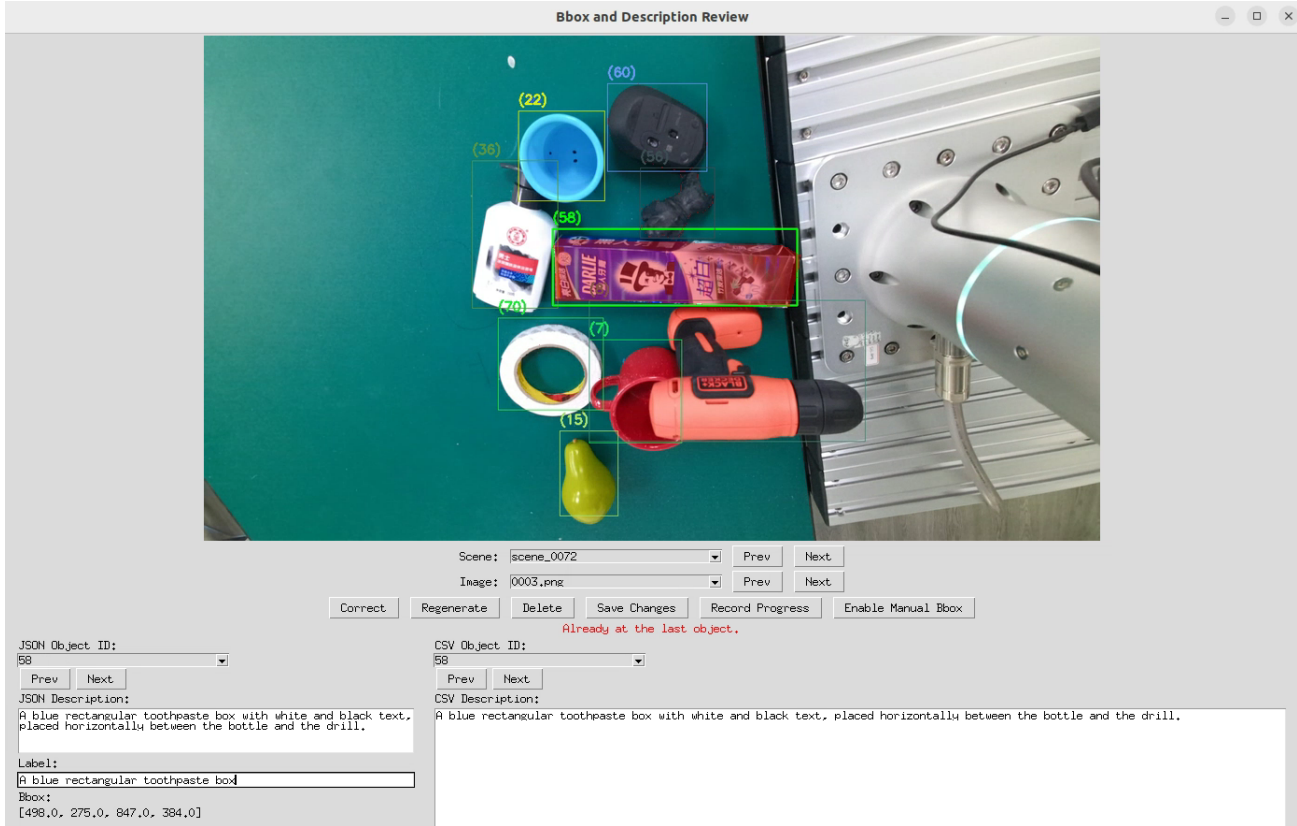


Figure 2. **Human-Verification System.** This application provides an interactive interface for human-in-the-loop verification, allowing users to review, correct, and confirm automatically generated visual-language annotations. It serves as a crucial component for ensuring the quality and reliability of RealVLG-11B dataset annotations.

#### Prompt 2.4: Bbox/Seg Task Prompt

Predict the bounding box of the referred object in the image based on the instruction: "`{{ content | trim }}`". First, output the thinking process in `<think>` `</think>` tags, then output the final answer in `<answer>` `</answer>` tags. Follow the format:

```
<think> thinking process
</think>
<answer>(x_min,y_min),
(x_max,y_max)</answer>
```

#### Prompt 2.5: Grasp Task Prompt

Predict a stable 2D rectangular grasp pose for the target object based on the instruction: "`{{ content | trim }}`". First, output your reasoning process in `<think>` `</think>` tags, then output the final grasp pose in `<answer>` `</answer>` tags. Follow the format:

```
<think> thinking process
</think>
<answer>(x, y, theta,
width)</answer>
```

### Prompt 2.6: Contact Task Prompt

Predict one stable two-finger grasp contact pair (two 2D coordinates) for the target object described in the instruction: "{ content | trim }". First, output the thinking process in `<think>` `</think>` tags, then output the final answer in `<answer>` `</answer>` tags. Follow the format:

```
<think> thinking process
</think>
<answer> (x1,y1), (x2,y2) </answer>
```

## 2.2. Details of RealVLG-R1

**Task Prompt.** In Fig. 3 of the main text, the prompts for the Bbox and Segmentation tasks are defined in Prompt 2.4, while the Grasp task and Contact task are defined in Prompt 2.5 and Prompt 2.6, respectively.

**Verifiable Rewards.** Each reward consists of two components: a *format reward*  $R_{\text{Format}}$  and a *task reward*  $R_{\text{Task}}$ . The format reward is obtained through regular expression matching, assigning  $R_{\text{Format}} = 1$  if the model output strictly follows the predefined format and  $R_{\text{Format}} = 0$  otherwise. The task reward is computed based on task-specific evaluation metrics and normalized to the range  $[0, 1]$ . The final composite reward is computed as a weighted sum:

$$R(q, o) = \alpha R_{\text{Format}} + \beta R_{\text{Task}}, \quad (5)$$

where  $\alpha = 0.1$  and  $\beta = 0.9$  balance structural validity and task-level accuracy.

## 3. Further Experiments

### 3.1. Qualitative Comparison of Data Quality

As illustrated in Fig. 3, we compare RealVLG-11B with the Grasp-Anything family of datasets in terms of image fidelity, linguistic specificity, and the reliability of grasp annotations. The language instructions in Grasp-Anything [7] primarily describe object categories, while Grasp-Anything++ [6] introduces limited part-level cues but remains restricted to highly templated and semantically simplistic expressions. Moreover, both datasets rely on diffusion-generated images with relatively low resolution ( $416 \times 416$ ), which exhibit noticeable artifacts and distortions, particularly in complex geometric structures and fine-grained textures. Their textual descriptions also show weak alignment with visual content, failing to consistently correspond to specific object instances in the scene.

In terms of grasp supervision, the Grasp-Anything datasets depend on RAGT-3/3 to synthesize grasp poses, which results in high-noise, low-precision annotations that lack semantic coherence with the associated language instructions. Such limitations make it challenging to support fine-grained visual-language-action learning.

In contrast, RealVLG-11B is constructed entirely from high-resolution real-world images ( $1280 \times 720$ ), preserving authentic geometric details, texture richness, and environmental variability, thereby enhancing both data realism and downstream generalization. Its linguistic annotations are produced by large vision-language models and subsequently validated by human experts, enabling instance-level grounding of objects and object parts and ensuring high-quality image-language alignment. Meanwhile, the grasp pose annotations in RealVLG-11B undergo standardized processing, yielding higher physical executability and stronger annotation accuracy. Overall, RealVLG-11B substantially surpasses the Grasp-Anything datasets in visual quality, semantic granularity, and grasp annotation reliability, providing a more robust foundation for training multi-modal robotic agents capable of grounded perception and stable manipulation in real-world environments.

### 3.2. Details of Baselines

For Gemini2.5-Flash, since the model can directly output segmentation masks, the evaluation for detection and segmentation tasks follows the prompt specified in its official documentation<sup>1</sup> (see Prompt 3.1).

### Prompt 3.1: Gemini2.5-Flash Evaluation Prompt

Give the segmentation masks for the object: "{ description }". Output a JSON list of segmentation masks where each entry contains the 2D bounding box in the key "box\_2d", the segmentation mask in key "mask", and the text label in the key "label". Use descriptive labels.

For other tasks, all baseline models use the same task prompt as RealVLG-R1.

### 3.3. Comparison of Optimization Strategies

Fig. 4 presents a comparative analysis of the training performance of GRPO, GSPO, and SFT on the grasp contact point prediction task, evaluated across 3B and 7B model scales. Overall, both GRPO and GSPO substantially outperform the SFT baseline by leveraging the Reward-Driven Learning with Verifiable Rewards (RLVR) paradigm, demonstrating its effectiveness in achieving precise visual grounding and grasping. Both reinforcement learning methods employ a group-wise advantage estimation strategy, which contributes to their early performance gains.

When considering model scale and optimization strategies, notable differences emerge. For the 3B models, GRPO exhibits a slight advantage due to its token-level importance weights, which provide finer-grained gradient

<sup>1</sup>[https://ai.google.dev/gemini-api/docs/image-understanding#python\\_5](https://ai.google.dev/gemini-api/docs/image-understanding#python_5)



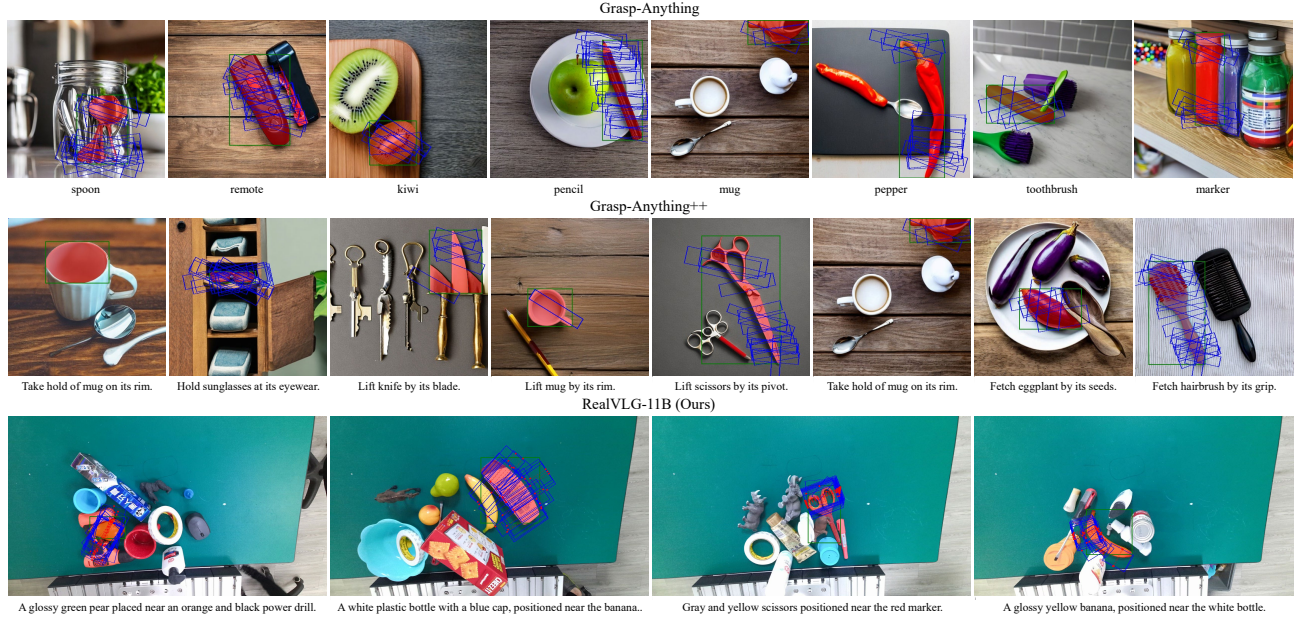


Figure 3. **Qualitative Comparison of Data Quality.** Unlike the diffusion-generated, low-resolution images and weakly aligned textual and grasp annotations in Grasp-Anything datasets, RealVLG-11B provides high-resolution real-world imagery, instance-level language grounding, and standardized, physically executable grasp labels, enabling more accurate and robust visual-language grasping.



Figure 4. **Training reward/accuracy curves for GRPO, GSPO, and SFT on Contact tasks.** Overall, GRPO and GSPO significantly improve SFT through RLVR. GRPO achieves slightly higher accuracy on 3B, while GSPO performs better on 7B and exhibits more stable outputs across training steps.

updates. This granularity facilitates more focused optimization in parameter-limited small models, resulting in marginally higher peak accuracy. In contrast, for the 7B models, GSPO outperforms GRPO by utilizing sequence-level importance weights with length normalization. This approach mitigates gradient variance over long sequences, enabling smoother optimization and allowing larger models to fully exploit their expressive capacity.

In terms of training stability, GSPO demonstrates

more consistent and stable convergence, whereas GRPO’s token-level weighting can induce higher variance in long-sequence tasks, leading to more oscillatory training behavior. These findings confirm the efficacy of RLVR in enhancing grasp contact point prediction and highlight a subtle trade-off between optimization granularity and model scale: fine-grained token-level optimization is more suitable for small models aiming for peak performance, while sequence-level optimization ensures robustness and convergence quality for large models.

## 4. Evaluation on Real Robot

### 4.1. Implementation Details

To assess the open-world generalization capability of our model in real-world environments, we conducted a series of robotic manipulation experiments. As illustrated in Fig. 5, the evaluation was performed using a 7-DoF Franka Research 3 manipulator equipped with an eye-in-hand Intel RealSense D435i RGB-D camera. We designed a suite of ten grasping tasks involving diverse household objects, including *Cup*, *Orange*, *Apple*, *Pear*, *Stapler*, *Blue bottle*, *Banana*, *Marker*, *Screwdriver*, and *Razor*. Each task was executed 10 times, and the average success rate was reported.

### 4.2. Experimental Settings

To comprehensively evaluate the generalization capability of our proposed RealVLG-R1 model in real-world environments, we designed experiments across two comple-

Method	Cup	Orange	Pear	Apple	Banana	Stapler	BlueBottle	Marker	Screwdriver	Razor	Average
GraspNet [2]	10%	60%	70%	60%	30%	50%	40%	20%	30%	10%	38%
RealVLG-R1 (Ours)	<b>90%</b>	<b>100%</b>	<b>100%</b>	<b>100%</b>	<b>100%</b>	<b>80%</b>	<b>70%</b>	<b>80%</b>	<b>100%</b>	<b>90%</b>	<b>81%</b>

Table 1. **Quantitative real-world grasping results in the Single setting.** RealVLG-R1 performs language-conditioned grasping, whereas GraspNet serves as a vision-only baseline. Each task is executed 10 times.

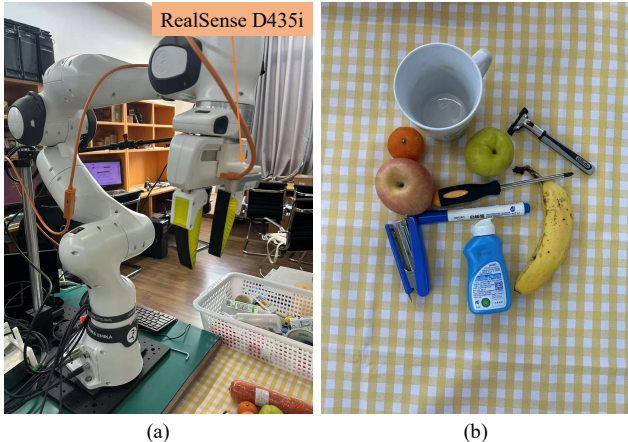


Figure 5. **Real-world experimental setup.** (a) The 7-DoF Franka Research 3 robot equipped with an eye-in-hand Intel RealSense D435i camera, used for real-world evaluation of RealVLG-R1. (b) The set of 10 test objects used to assess the model’s generalization and manipulation performance.

mentary settings and selected two representative baselines for comparison. In the *Single* setting, only the target object is placed to ensure stable operation of the classical geometric grasping baseline GraspNet [2], which lacks language-conditioned capabilities. This setting aims to assess RealVLG-R1’s performance in basic geometric grasping accuracy. In the more challenging *Clutter* setting, all 10 objects are placed simultaneously, and the robot is required to sequentially grasp the specified targets according to language instructions. In this scenario, RealVLG-R1 is compared only with LGD [6], a baseline that supports language-conditioned grasping. **The comparison is conducted using the RealVLG-R1 3B model trained with GRPO, which predicts grasp contact points.** This hierarchical evaluation strategy allows us to clearly demonstrate RealVLG-R1’s semantic reasoning advantages over purely geometric baselines, while highlighting its zero-shot deployment performance and cross-modal generalization capability in highly complex cluttered scenes.

### 4.3. Experimental Results

**Grasping Results in the Single Setting.** As shown in Table 1 and Fig. 6, RealVLG-R1 exhibits a substantial performance advantage over GraspNet in the Single setting. Although GraspNet is capable of predicting 6-DoF grasp

poses, its effectiveness in real-world scenarios is heavily constrained by the quality of the reconstructed point cloud. When objects contain reflective surfaces, sparse textures, or complex geometries, the depth measurements often become incomplete or noisy. For instance, for objects such as the *Cup*, the material and surface properties frequently lead to partial or degraded point cloud reconstructions, preventing GraspNet from producing any feasible grasp pose. Furthermore, for small and slender objects (such as the *Marker*, *Screwdriver*, and *Razor*), their point clouds tend to merge with the tabletop during reconstruction, making them nearly indistinguishable in the depth domain. As a result, GraspNet commonly predicts grasp poses that fall outside the true object region, ultimately causing grasp failures.

In contrast, RealVLG-R1 relies solely on RGB visual information and leverages language-conditioned grounding to accurately localize the target object. The model predicts grasp contact points directly in the image domain, and as long as the depth along the line connecting these contact points is valid, a physically executable grasp can be produced. This design effectively circumvents the depth degradation issues that commonly arise in real-world sensing (such as missing geometry, noise, or surface ambiguities), allowing RealVLG-R1 to maintain robust grasp performance across diverse visual conditions.

Overall, RealVLG-R1 achieves a significantly higher mean success rate than GraspNet in real-world single-object grasping, yielding a 43% improvement, which underscores the robustness and practical effectiveness of a vision-based, language-driven grasping framework in real-world environments.

**Grasping Results in the Clutter Setting.** As shown in Table 2, in the highly challenging Clutter Setting, our proposed RealVLG-R1 demonstrates overwhelming performance advantages. The model achieves a grasp success rate of 70% or higher across all 10 target objects, with an average success rate of 79%. These results strongly indicate that RealVLG-R1 is capable of accurately identifying, localizing, and executing grasping tasks according to natural language instructions, even in environments characterized by high object density and cluttered interference. Although certain grasp contact points may occasionally exhibit instability, suggesting that further enhancement is needed for real-world deployment, the overall performance clearly demonstrates the model’s strong zero-shot generalization capability.



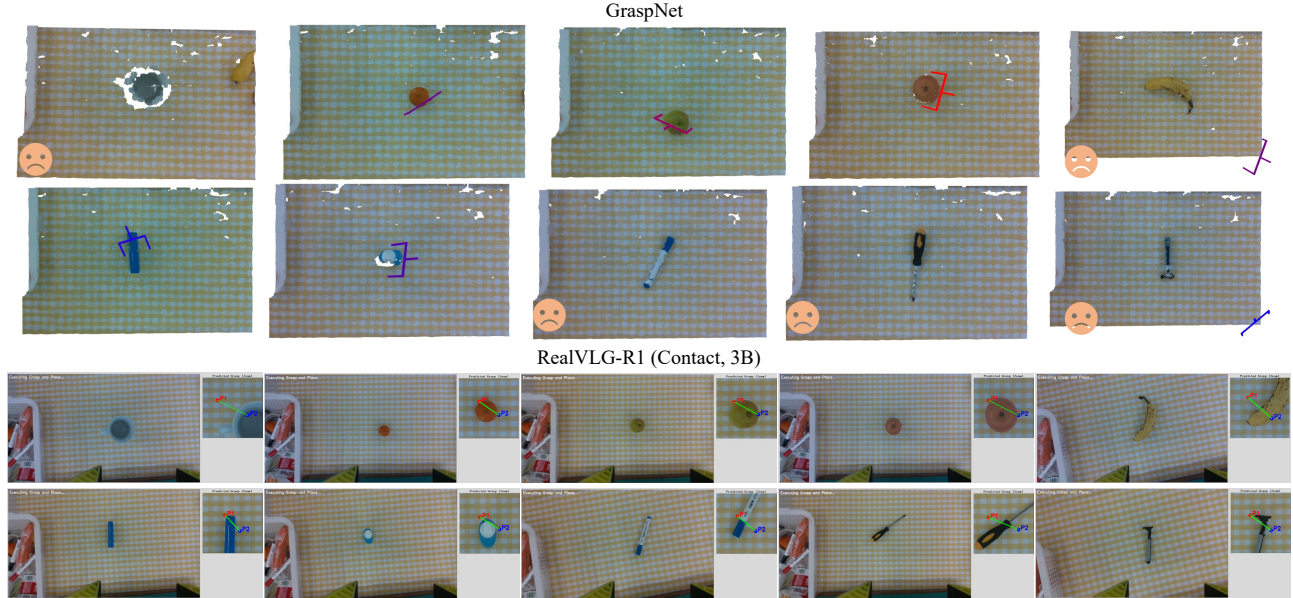


Figure 6. **Qualitative real-world grasping results in the Single setting.** GraspNet often fails or predicts misaligned grasp poses due to noisy or incomplete point cloud data (e.g., *Cup*), reflective surfaces, and small or thin objects, such as *Marker*, *Screwdriver*, and *Razor*. In contrast, RealVLG-R1 leverages RGB vision and language instructions to accurately localize the target and generate executable grasp contact points, demonstrating robust and reliable grasping behavior across diverse objects.

Method	Cup	Orange	Pear	Apple	Banana	Stapler	BlueBottle	Marker	Screwdriver	Razor	Average
LGD [6]	0%	0%	0%	0%	0%	20%	0%	0%	0%	0%	2%
RealVLG-R1 (Ours)	<b>70%</b>	<b>90%</b>	<b>80%</b>	<b>90%</b>	<b>70%</b>	<b>90%</b>	<b>70%</b>	<b>80%</b>	<b>70%</b>	<b>80%</b>	<b>79%</b>

Table 2. **Quantitative real-world grasping results in the Clutter setting.** The table reports the grasp success rates of RealVLG-R1 and the baseline LGD across 10 target objects. Results highlight RealVLG-R1’s superior zero-shot performance in cluttered multi-object environments, whereas LGD fails to reliably perform language-conditioned grasping.

In contrast, the baseline LGD achieves an average success rate of only 2% in real-world cluttered scenes, highlighting its failure in language-conditioned grasping tasks. This significant gap arises from inherent design limitations and training data constraints: the network, based on the GGCNN [3] architecture, processes images of only  $224 \times 224$  pixels, resulting in a severely restricted perceptual field that hampers accurate recognition of object context and complete spatial information in densely cluttered environments. Although LGD attempts to incorporate language instructions through CLIP [5] embeddings, experimental results indicate that such integration is suboptimal. As illustrated in Fig. 7, LGD rarely predicts grasp poses corresponding to specific objects based on language instructions, with its output grasp poses largely independent of the provided commands. Furthermore, the “accuracy” reported for LGD on Grasp-Anything++ dataset predominantly reflects the unconditional grasp pose prediction capabilities inherited from GGCNN, rather than poses specified by language instructions. In the absence of high-quality language-to-pose aligned training data, LGD’s language-driven object

grasping success in real-world scenes is effectively negligible, with the few successful cases (e.g., a 20% success rate for the *Stapler*) attributable to geometric cues alone.

In sharp contrast to LGD’s limitations, RealVLG-R1 exhibits precise and reliable language-conditioned grasping capability, enabling the model to accurately localize target objects in wide-field, multi-object scenarios. Moreover, as shown in Fig. 8, RealVLG-R1 provides strong reasoning capabilities, offering compelling interpretability for predicted grasp poses. This reasoning mechanism not only enhances model transparency and reliability but also provides an additional internal validation of grasp pose accuracy. By combining precise perception, accurate language alignment, and interpretable reasoning, RealVLG-R1 achieves high grasping success rates in cluttered environments, fully demonstrating its zero-shot deployment performance and cross-modal generalization capabilities in complex real-world scenarios.

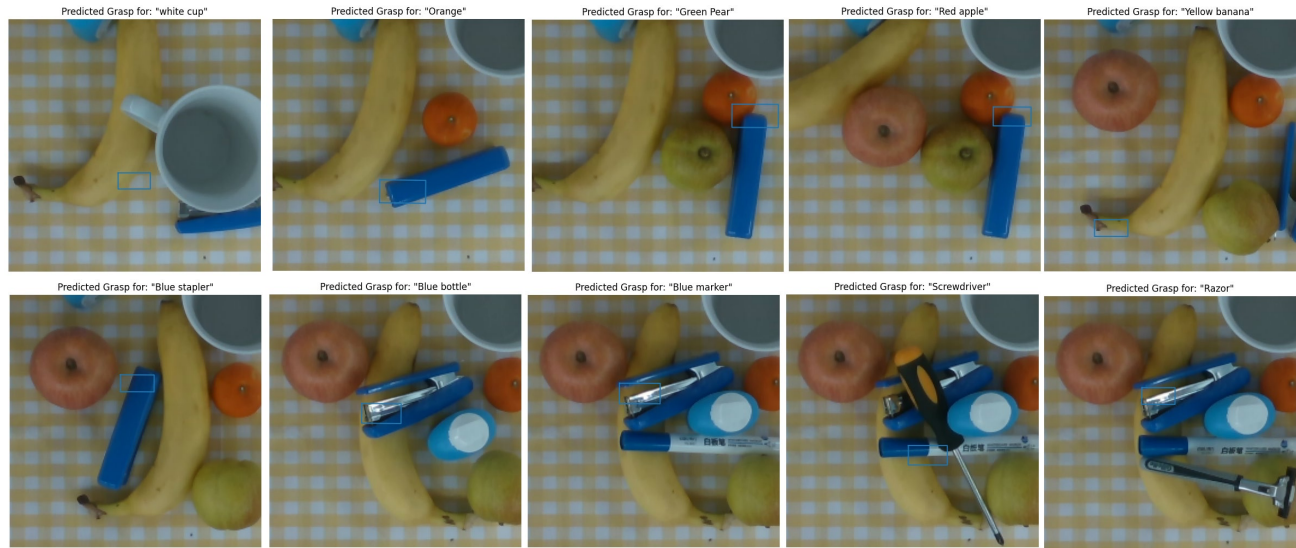


Figure 7. **Qualitative real-world grasping results of LGD [6] in the Clutter setting.** LGD struggles to perform language-conditioned grasps in cluttered environments due to limited perceptual resolution, suboptimal language integration, and reliance on unconditional grasp pose predictions.

## References

- [1] Alibaba Cloud. Qwen-VL-Max: Alibaba’s large visual language model. [https://modelstudio.console.alibabacloud.com/?tab=doc#/doc/?type=model&url=2840914\\_2&modelId=qwen-vl-max](https://modelstudio.console.alibabacloud.com/?tab=doc#/doc/?type=model&url=2840914_2&modelId=qwen-vl-max), 2025. Accessed: 2025-11-06. 2
- [2] Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. GraspNet-1Billion: A large-scale benchmark for general object grasping. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 11441–11450, 2020. 1, 6
- [3] Douglas Morrison, Peter Corke, and Jürgen Leitner. Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach. In *RSS*, 2018. 7
- [4] OpenAI. GPT-4o: OpenAI’s multimodal large language model. <https://openai.com/index/hello-gpt-4o>, 2024. Accessed: 2025-11-06. 2
- [5] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Int. Conf. Mach. Learn.*, pages 8748–8763, 2021. 7
- [6] An Dinh Vuong, Minh Nhat Vu, Baoru Huang, Nghia Nguyen, Hieu Le, Thieu Vo, and Anh Nguyen. Language-driven grasp detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 17902–17912, 2024. 4, 6, 7, 8
- [7] A. D. Vuong, M. N. Vu, H. Le, B. Huang, H. T. T. Binh, T. Vo, A. Kugi, and A. Nguyen. Grasp-Anything: Large-scale grasp dataset from foundation models. In *IEEE Int. Conf. Robot. Autom.*, pages 14030–14037, 2024. 4



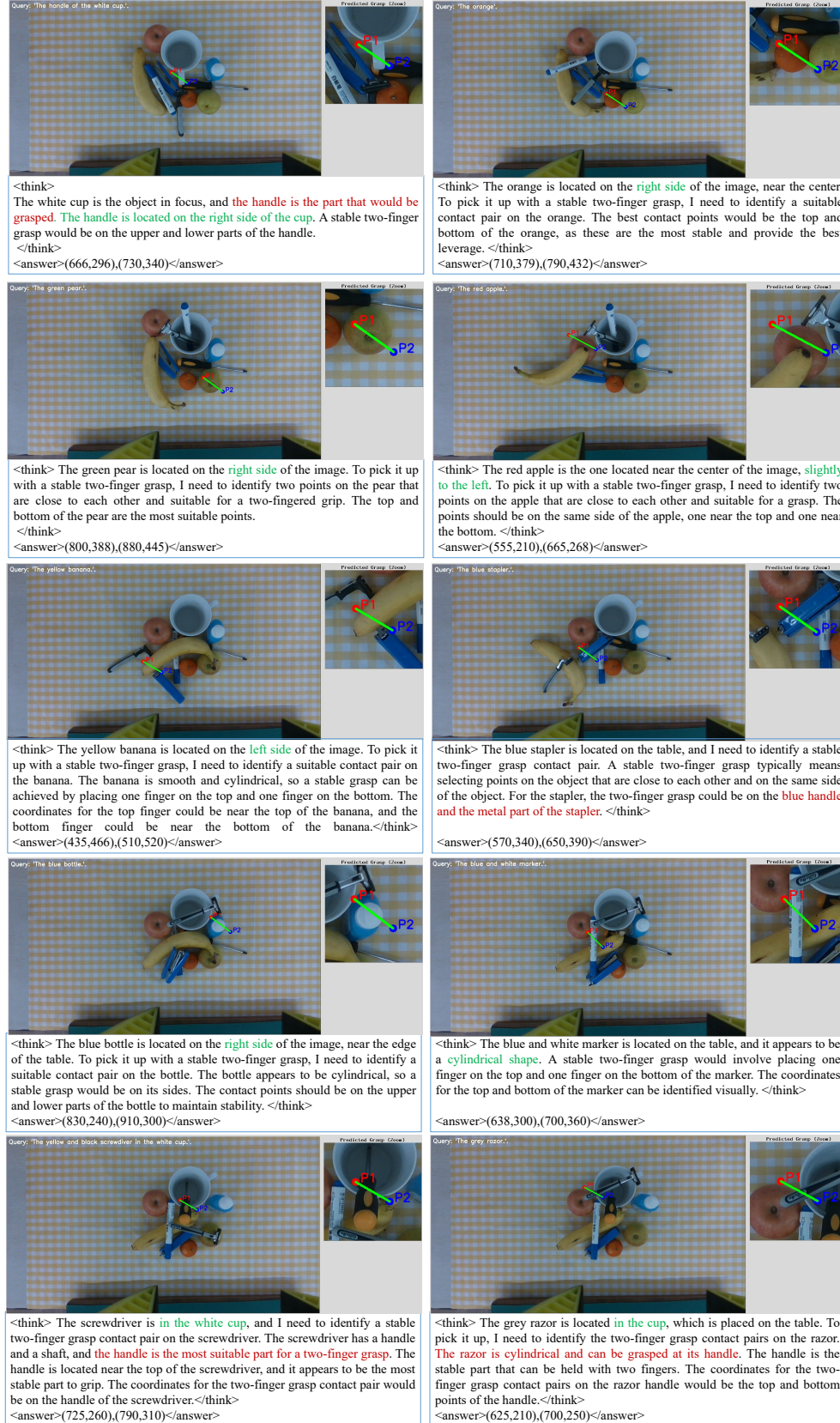


Figure 8. **Qualitative real-world grasping results of RealVLG-R1 in the Clutter setting.** RealVLG-R1 demonstrates accurate language-conditioned grasping, robust zero-shot performance in cluttered environments, and interpretable predictions of grasp poses.