GREAT: Geometry-Intention Collaborative Inference for Open-Vocabulary 3D Object Affordance Grounding

Supplementary Material

Contents

A. Implementation Details **1**
- A.1. Method Details 1
- A.2. Evaluation Metrics 1
- A.3. Training Details 2

B. Dataset **2**

C. Experiments **3**
- C.1. Details of Modular Baselines 3
- C.2. Detailed Metric Results 4
- C.3. More Visual Results 4
- C.4. Partial and Rotated Results 4
- C.5. Computational Complexity 5
- C.6. More comparative experiments 5
- C.7. Application in robotics 5

A. Implementation Details

A.1. Method Details

We demonstrate dimensions and meanings of tensors in the GREAT pipeline as shown in Tab. 1. For the image branch, ResNet18 [4] is chosen as the feature extractor. The input image is randomly cropped and resized to 224 × 224, producing image features with a shape of $\mathbf{F}_i \in \mathbb{R}^{512\times7\times7}$. A 1×1 convolutional layer is applied to reduce the feature dimension and the feature is flattened to $\mathbf{F}_i \in \mathbb{R}^{512\times49}$. For the point branch, each input point cloud contains 2048 points. We employ pointnet++ [11], which consists of three set abstraction (SA) layers, to progressively extract multi-scale point cloud features. Within each SA layer, Farthest Point Strategy (FPS) is used to sample points, with the sampling counts set to 512, 128, and 64. Ultimately, this branch outputs point features represented as $\mathbf{F}_p \in \mathbb{R}^{512\times2048}$. Detailed prompts on Multi-Head Affordance Chain-of-Thought (MHACoT) reasoning are presented below.

— **Prompt One**: "*Point out which part of the object in the image interacts with the person. If this part is different from the part of the object shown in the image that performs the main function, point out the part of the object that performs the main function shown in the image.*"

— **Prompt Two**: "*Explain why this part can interact from the geometric structure of the object. Just give the final result in one sentence.*"

— **Prompt Three**: "*Describe the interaction between object and the person in the image, including the interaction type, the interaction part of the object, and the interaction part of the person.* "

— **Prompt Four**: "*List two interactions that describe additional common interactions that the object can interact with people, including the interaction type, the interaction part of the object, and the interaction part of the person.* "

We connect the answers of Prompt One and Prompt Two, as well as the answers of Prompt Three and Prompt Four, to obtain object geometric knowledge feature $\mathbf{T}_o \in \mathbb{R}^{1\times512}$ and affordance intention knowledge feature $\mathbf{T}_a \in \mathbb{R}^{3\times512}$ through the text encoder RoBERTa [8].

Table 1. **Tensors.** The dimension and meaning of the tensors in the pipeline.

Tensor	Dimension	Meaning
$\mathbf{F_i}$	$512 \times 7 \times 7$	image extractor output
$\mathbf{F}_p$	512×64	point cloud extractor output
$\mathbf{T}_o, \bar{\mathbf{T}}_o$	1×512	object geometric knowledge feature
$\mathbf{T}_a, \bar{\mathbf{T}}_a$	3×512	affordance intention knowledge feature
$\mathbf{F}_p^{'}$	512×64	project $\mathbf{F}_p$ to a feature space
$\bar{\mathbf{T}}_o^{'}$	512×1	project $\bar{\mathbf{T}}_o$ to a feature space
$\mathbf{P}_o$	512×64	fused point features by $\mathbf{F}_p$,$\bar{\mathbf{T}}_o$
$\mathbf{F}_{tp}$	512×2048	upsampled fused point features by $\mathbf{P}_o$
$\mathbf{F}_{ti}$	512×16	fused image features by $\mathbf{F}_i$,$\bar{\mathbf{T}}_a$
$\mathbf{F}_\alpha$	512×2048	affordance feature representation
ϕ	2048×1	3D object affordance

A.2. Evaluation Metrics

We employ four evaluation metrics to assess performance: **AUC** [9], **aIOU** [12], **SIM** [13], and **MAE** [14]. A detailed explanation of each metric is provided below:

— **AUC** [9]: AUC is a widely adopted metric for evaluating saliency maps, treating them as binary classifiers across varying thresholds. By computing the true and false positive rate at each threshold, it produces the ROC curve, which captures the model's classification performance. In our work, AUC is utilized to evaluate the model's capability to differentiate between affordance and non-affordance regions of an object with 2048 points.

— **aIOU** [12]: IOU is a critical metric for assessing the similarity between two regions, widely employed to quantify the degree of overlap between predicted and ground truth regions. Its range is $[0, 1]$, where 1 indicates perfect overlap and 0 signifies no intersection. IOU is defined as the ratio of the intersection area to the union area of the two

regions, formulated as:

$$\text{IOU} = \frac{\text{Intersection Area}}{\text{Union Area}}, \tag{1}$$

The aIOU is defined as the mean IOU value computed over multiple thresholds, formulated as:

$$\text{aIOU} = \frac{1}{T}\sum_{i=1}^{T}\text{IOU}_i, \tag{2}$$

where T denotes the number of thresholds.

– **SIM** [13]: SIM measures the similarity between the prediction map (P) and the ground truth map (Q^D), formulated as:

$$\begin{aligned} &SIM(P, Q^D) = \sum_i min(P_i, Q_i^D), \\ &where \quad \sum_i P_i = \sum_i Q_i^D = 1. \end{aligned} \tag{3}$$

– **MAE** [14]: MAE is a widely used metric for evaluating models, quantifying the deviation between predicted and true values. It is calculated by averaging the absolute differences between the predicted values and the corresponding true values, as formulated as dividing the total error by N:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|y_i - \hat{y}_i\right|, \tag{4}$$

where y_i denotes the ground truth, $\hat{y}_i$ denotes the prediction.

A.3. Training Details

In the fine-tuning process of MLLM, we exclusively fine-tune the parameters of the injected learnable adapters [5], while freezing the primary parameters of the InternVL [2]. The training is conducted out on two NVIDIA 3090 GPUs, using a dataset of 7135 samples that are unseen in the test sets across three different data settings. The model is trained for 4460 iterations with a learning rate of 4e-5, a batch size of 4, and a LoRA rank set to 16.

To ensure a fair comparison, we train our model and implement all baseline methods under identical training settings. Our model is built using the PyTorch framework and optimized with the Adam [6] optimizer. The training epoch is set to 65, with an initial learning rate of 1e-4 and a batch size of 16. All training processes are conducted on two NVIDIA 3090 GPUs. The image feature extractor leverages pretrained parameters from ImageNet, the parameters of text feature extractor are frozen, while the point cloud feature extractor is trained from scratch. Furthermore, as strict one-to-one pairing between images and point clouds is not required, we adopt an online pairing strategy during training. In each training step, a single image can be paired with n point clouds, effectively augmenting the training sample size. Considering both training efficiency and model performance, we set $n = 2$ in our implementation to strike an optimal balance.

Table 2. **Unseen Objects.** The affordance and corresponding number of images and point clouds for each object in the test set under the **Unseen Object** setting.

Object	Affordance	Image	Point
Scissors	Cut, Grasp, Stab	130	410
Baseballbat	Wrapgrasp	516	112
Mop	Wrapgrasp, Clean	286	17
Clock	Display	143	1009
Refrigerator	Contain, Open	147	290
Bucket	Contain, Lift	107	234
Motorcycle	Ride	486	301
Fork	Wrapgrasp, Stab	240	90
Skateboard	Support	641	152
Laptop	Display, Press	296	679
Kettle	Contain, Grasp, Open, Pour	280	524

Table 3. **Unseen Affordances.** The object and corresponding number of images and point clouds for each affordance in the test set under the **Unseen Affordance** setting.

Affordance	Object	Image	Point
Cut	Scissors, Knife	366	425
Pour	Bottle, Kettle, TrashCan, Mug	435	2945
Pull	Suitcase	181	20
Lay	Bed	289	779
Carry	Backpack, Surfboard	377	118
Listen	Earphone	365	710

B. Dataset

We provide a detailed description of the dataset partitioning process. PIADv2 consists of 43 object categories and 24 affordance categories. To validate the effectiveness of GREAT for object affordance grounding in an open-vocabulary scenario, we divide the dataset into three partitions: **Seen**, **Unseen Object** and **Unseen Affordance**. In **Seen**, all object and affordance categories in the test set are identical to those in the training set. In **Unseen Object**, the affordances remain consistent with the training set, but several objects are excluded from the training set. The following eleven objects are selected as the test set for unseen object: "Scissors", "Baseballbat", "Mop", "Clock", "Refrigerator", "Bucket", "Motorcycle", "Fork", "Skateboard", "Laptop", "Kettle". The ratio of object categories between

the training set and the test set is 32:11. The affordance categories corresponding to each unseen object, along with the number of associated images and point clouds, are detailed in the Tab. 2. Notably, a fixed one-to-one correspondence is not required, as a single image can be paired with multiple point clouds. In **Unseen Affordance**, the certain affordances of object categories in the test set are not present in the training set. Specifically, the following six affordances are selected as the test set for unseen affordance: "Cut," "Pour", "Pull", "Lay", "Carry", "Listen". Notably, the earphone (which corresponds to the action "Listen") and the suitcase (which corresponds to the action "Pull") are also absent from the training set, further increasing the challenge for generalization. The ratio of affordace categories between the training set and the test set is 18:6. The object categories corresponding to each unseen affordance, along with the number of associated images and point clouds, are detailed in the Tab. 3.

C. Experiments

C.1. Details of Modular Baselines

We have selected two leading 3D object affordance grounding methods, IAG [16] and LASO [7], which leverage either the interaction image or the language guiding the interaction to obtain additional contextual information. In addition, we have chosen two of the top-performing image-point cloud cross-modal learning methods compared in IAG, FRCNN [15] and XMF [1] . These methods respectively extract features from image and point cloud data and align or fuse the extracted features. We reimplement the above four methods across three data settings in PIADv2, where all compared methods share the same feature extractor as our GREAT.

- **Baseline**: For the design of the baseline, we directly connect the features output by the image and point cloud extractors, and then use the output head to predict the affordance of 3D object point clouds, without any intermediate steps to align features from different sources.
- **FusionRCNN (FRCNN)** [15]: This work tackles the challenge of object recognition and localization caused by the sparsity of point clouds in distant regions, proposing a novel multi-modal two-stage approach. The method effectively integrates point cloud data and camera images in the region of interest (RoI), adaptively combining sparse LiDAR geometric information with dense camera texture information within a unified attention mechanism.
- **XMFnet (XMF)** [1]: This work explores the problem of point cloud completion using edge information provided by a single image and shape priors. By combining self-attention and cross-attention mechanisms, it effectively fuses features from two different modalities, integrating the information from both modalities into a local latent space. It avoids the complex point cloud reconstruction methods typically used in single-view techniques.
- **IAGNet (IAG)** [16]: This work leverages the human ability to perceive object affordance in the physical world through demonstration images. It proposes a method to locate 3D object affordance from 2D interactions in images, aligning region features of objects from different sources. Additionally, to resolve the ambiguity of affordance, the dynamic factors involved in affordance extraction are decomposed into interactions between the subject-object and object-scene. Contextual modeling of these interactions reveals explicit affordance.
- **LASO** [7]: This work explores the synergy with large language models (LLMs) and proposes the setting of language-guided 3D object affordance segmentation, aiming to segment 3D object affordance based on given expert-crafted questions. The method introduces an adaptive fusion module to identify target affordance regions at different scales and utilizes a set of affordance queries conditioned on linguistic clues to generate dynamic kernels. These dynamic kernels are then convolved with point cloud features to produce segmentation masks. We use the InternVL to generate question descriptions for the

Table 4. **Evaluation Metrics in Unseen Affordance.** Results of each affordance type for all comparison methods in the unseen affordance setting.

Setting	Metrics	Carry	Listen	Lay	Pour	Cut	Pull
Baseline	**AUC**	57.39	48.57	69.45	59.96	39.55	93.61
	aIOU	7.33	3.43	6.85	6.08	4.25	31.42
	SIM	0.237	0.152	0.324	0.184	0.105	0.348
	MAE	0.147	0.221	0.131	0.145	0.208	0.054
FRCNN [15]	**AUC**	52.63	50.40	75.68	59.54	43.84	93.28
	aIOU	5.39	3.23	10.70	6.17	4.62	29.58
	SIM	0.178	0.157	0.411	0.179	0.098	0.371
	MAE	0.17	0.20	0.12	0.14	0.20	0.04
XMF [1]	**AUC**	54.08	56.07	73.16	63.97	44.85	91.40
	aIOU	5.89	3.89	10.93	6.85	5.77	24.52
	SIM	0.195	0.216	0.399	0.187	0.115	0.349
	MAE	0.158	0.179	0.130	0.130	0.213	0.050
IAG [16]	**AUC**	63.98	54.13	69.94	59.89	49.97	93.72
	aIOU	8.10	3.71	10.47	4.92	4.40	38.27
	SIM	0.239	0.221	0.402	0.146	0.148	0.562
	MAE	0.142	0.168	0.130	0.146	0.175	0.028
LASO [7]	**AUC**	65.09	46.95	78.52	60.64	59.49	90.99
	aIOU	7.22	2.20	10.23	5.93	9.61	23.60
	SIM	0.262	0.116	0.404	0.152	0.196	0.350
	MAE	0.138	0.209	0.126	0.124	0.158	0.043
Ours	**AUC**	82.13	51.36	77.53	72.82	52.21	97.39
	aIOU	12.59	2.48	10.66	11.28	8.53	41.53
	SIM	0.356	0.125	0.412	0.290	0.143	0.599
	MAE	0.105	0.182	0.129	0.108	0.171	0.018

Table 5. **Evaluation Metrics in Unseen Object.** Results of each object type in the unseen object setting. "Base." denotes "Baseballbat", "Motor." denotes "Motorcycle", "Refri." denotes "Refrigerator", and "Scis." denotes "Scissors", "Skat." denotes "Skateboard".

Method	Metrics	Base.	Bucket	Clock	Fork	Kettle	Laptop	Mop	Motor.	Refri.	Scis.	Skat.
Baseline	**AUC**	69.28	51.67	75.62	79.28	60.89	74.15	85.81	76.78	82.72	57.15	72.47
	aIOU	36.03	6.17	20.63	26.36	4.75	8.19	27.21	6.90	12.43	9.98	10.81
	SIM	0.502	0.104	0.437	0.471	0.085	0.262	0.481	0.162	0.349	0.255	0.396
	MAE	0.214	0.198	0.141	0.139	0.135	0.205	0.141	0.064	0.089	0.169	0.192
FRCNN [15]	**AUC**	76.40	57.24	82.14	83.09	41.29	72.25	90.13	59.18	82.06	64.49	79.72
	aIOU	41.76	6.83	24.20	29.01	2.35	8.73	30.24	3.52	6.98	11.17	16.23
	SIM	0.572	0.125	0.500	0.512	0.054	0.293	0.549	0.099	0.294	0.289	0.459
	MAE	0.171	0.190	0.133	0.134	0.174	0.169	0.119	0.084	0.128	0.168	0.197
XMF [1]	**AUC**	76.75	59.73	77.86	81.93	61.74	73.52	79.09	69.02	84.22	56.31	81.80
	aIOU	37.62	10.26	19.13	27.23	5.50	8.66	20.53	5.54	9.41	10.24	18.38
	SIM	0.528	0.169	0.444	0.497	0.107	0.314	0.400	0.162	0.336	0.261	0.482
	MAE	0.171	0.165	0.137	0.120	0.123	0.128	0.136	0.033	0.094	0.165	0.151
IAG [16]	**AUC**	85.32	65.52	69.21	73.63	54.38	78.17	92.09	79.74	80.63	56.62	58.84
	aIOU	42.84	12.56	13.48	22.60	2.65	6.95	37.66	9.30	14.43	7.37	4.61
	SIM	0.592	0.231	0.385	0.422	0.070	0.299	0.658	0.227	0.335	0.250	0.273
	MAE	0.169	0.146	0.148	0.149	0.120	0.104	0.085	0.023	0.092	0.187	0.167
LASO [7]	**AUC**	73.99	52.61	80.64	85.83	63.68	69.20	87.21	72.21	72.70	58.26	73.91
	aIOU	38.58	6.51	22.12	30.63	4.92	6.33	28.05	4.43	4.60	9.38	9.52
	SIM	0.551	0.148	0.477	0.570	0.093	0.259	0.590	0.138	0.179	0.288	0.394
	MAE	0.184	0.160	0.113	0.087	0.125	0.137	0.100	0.030	0.081	0.164	0.162
Ours	**AUC**	82.73	83.41	84.00	89.28	79.33	74.98	91.80	95.73	78.69	64.83	59.50
	aIOU	41.93	28.75	20.23	31.83	7.33	5.17	32.37	17.79	13.45	12.97	8.72
	SIM	0.584	0.469	0.486	0.567	0.182	0.242	0.612	0.356	0.294	0.283	0.324
	MAE	0.148	0.081	0.111	0.135	0.048	0.130	0.104	0.033	0.073	0.159	0.149

input interactive images.

C.2. Detailed Metric Results

To thoroughly evaluate the performance of the GREAT model, we present the results for each affordance or object category under the three data settings. Compared to other methods, our approach achieves optimal results across the majority of affordance or object categories. For the unseen affordance setting, we present results for each affordance category that does not appear in the training set (Tab. 4). For the unseen object setting, we provide results for each object category that does not appear in the training set (Tab. 5). While for the seen setting, we present results for each affordance category (Tab. 10). The experimental results demonstrate that, under the same task settings, our model exhibits strong robustness and excellent generalization capabilities, indirectly validating the rationality of the open-vocabulary 3D object affordance grounding task setting.

C.3. More Visual Results

We present the visualization results of GREAT on three different partitions. Fig. 1 shows the results for seen setting, while Fig. 2 presents the results for unseen object and unseen affordance settings. The results demonstrate that GREAT can accurately predict the affordance regions of 3D object in diverse interactive images and across multiple object categories, highlighting its stability, robustness, and generalization ability.

C.4. Partial and Rotated Results

Following the experimental setting proposed in [3] and [16], we tested the model's performance on partial and freely rotated point clouds, simulating object occlusion and rotation in daily environments. The corresponding visualization results are shown in the Fig. 3 and Fig. 4. The results demonstrate that even when the point clouds contain only partial object structures or are randomly rotated in space, our model can still accurately predict the 3D affordance of the object in open-vocabulary scenarios. This indicates that the reasoning knowledge from 2D interactions provides crucial cues that help the model understand the correlation between geometric structure and affordance. This capability of the model provides strong support for robots to quickly adapt to a wide range of real-world scenarios and respond to changes in the operational environment, making it highly effective for dynamic, real-world tasks.

Table 6. **Comparison on the PIAD.** Evaluation metrics of comparison methods on the PIAD benchmark, ◇ denotes the relative improvement of our method over IAG method.

	Seen				Unseen Object				Unseen Affordance			
Methods	**AUC ↑**	**aIOU ↑**	**SIM ↑**	**MAE↓**	**AUC ↑**	**aIOU ↑**	**SIM ↑**	**MAE↓**	**AUC ↑**	**aIOU ↑**	**SIM ↑**	**MAE↓**
IAG [16]	82.88	18.88	0.544	0.098	68.49 ◇1.3%	7.22 ◇3.7%	0.344 ◇2.3%	0.139 ◇8.6%	55.36 ◇13.1%	6.50 ◇0.9%	0.203 ◇30.1%	0.170 ◇15.9%
Ours	85.22	19.61	0.569	0.093	69.41	7.49	0.352	0.127	62.59	6.56	0.264	0.143

Table 7. **Comparison with OpenAD.** Evaluation metrics of comparison methods on the PIADv2 benchmark, ◇ denotes the relative improvement of our method over OpenAD method.

	Seen				Unseen Object				Unseen Affordance			
Methods	**AUC ↑**	**aIOU ↑**	**SIM ↑**	**MAE↓**	**AUC ↑**	**aIOU ↑**	**SIM ↑**	**MAE↓**	**AUC ↑**	**aIOU ↑**	**SIM ↑**	**MAE↓**
OpenAD [10]	89.54	31.88	0.526	0.104	73.49 ◇8.3%	16.62 ◇21.3%	0.339 ◇18.6%	0.159 ◇31.5%	61.22 ◇14.0%	8.00 ◇50.6%	0.229 ◇26.6%	0.167 ◇23.9%
Ours	91.99	38.03	0.676	0.067	79.57	20.16	0.402	0.109	69.81	12.05	0.290	0.127

C.5. Computational Complexity

The comparison results of the computational complexity metrics are presented in the Tab. 8, including model inference time (MIT.), chain-of-thought inference time (CoT-IT.), model parameters and trainable parameters. Although the CoT-IT. is slightly longer, it results in performance gains (Tab. 2 main paper), a trend also observed in LLMs like GPT-O1 and DeepSeek-R1. We mitigate the real-time reasoning burden during training by pre-generating and storing the CoT reasoning knowledge base, thereby significantly reducing the computation overhead and training complexity. Besides, fine-tuning MLLM only contains 25M trainable parameters.

C.6. More comparative experiments

Benchmark fairness. To ensure the fairness of the benchmark, we train our method on PIAD and compare it to IAG[16], as shown in Tab. 6. It is worth emphasizing that all compared methods in Tab. 2 main paper are re-trained on PIADv2. Our method shows excellent performance on both PIAD and PIADv2.

Comparison with clip-based method. To clearly motivate the needing of a big MLLM with respect to more compact text-models, we compare with the OpenAD[10] method on PIADv2, as shown in Tab. 7.

Combine two knowledge through one encoder. At the level of method design, is it effective to combine "object geometric properties" and "affordance interaction intentions" through an encoder? For example, we combine them through a text encoder in the unseen object partition, resulting in performance degradation, as shown in Tab. 9. Geometric attributes focus on object physical shape, while interaction intentions focus on the functionality of an object in a specific context. Forcing both into one encoder may over-couple, resulting in confusing information and making it difficult to capture the respective semantic features.

Table 8. **Computational Complexity.** MIT.: model inference time. CoT-IT.: chain-of-thought inference time. Model Params.: model parameters. Trainable Params.: trainable parameters.

Method	**MIT.**	**CoT-IT.**	**Model Params.**	**Trainable Params.**
IAG [16]	1.426s	–	24.7M	24.7M
LASO [7]	1.336s	–	130.5M	130.5M
Ours	1.272s	6.086s	256.7M	23.1M

Table 9. **Combine two knowledge through one encoder.** Evaluation metrics of our method with one encoder on the PIADv2 benchmark, ◇ denotes the relative improvement of our method over the method with one encoder.

AUC ↑	**aIOU ↑**	**SIM ↑**	**MAE ↓**
73.03◇8.9%	17.95◇12.3%	0.384◇4.7%	0.113◇3.5%

C.7. Application in robotics

We provide an explanation of how the proposed method can be effectively applied to real robots.The training process of the method involves establishing a mapping between 2D interaction contents and 3D object regions, which is not limited to a specific instance. Once this mapping is established, the input 2D interaction contents can be easily obtained in various ways *e.g.*, constructing a knowledge base or leveraging the large generative model like Stable Diffusion 3. To build such a mapping that can generalize across instances, the input image and point cloud keep a multi-to-multi pairing during training. This allows the model to learn similar geometries of distinct instances, thereby improving the model's ability to generalize to real-world applications. Although the interaction methods may be different, the interaction regions of objects are mostly consistent, through the above manner, we enable the model to build this consistency.

Table 10. **Evaluation Metrics in Seen.** Results of each affordance type for all comparison methods in the seen setting. "Cont." denotes "Contain", "Supp." denotes "Support", "Wrap." denotes "Wrapgrasp", and "Disp." denotes "Display".

Method	Metrics	Grasp	Cont.	Lift	Open	Lay	Sit	Supp.	Wrap.	Pour	Move	Disp.	Push
Baseline	**AUC**	78.94	67.18	91.89	84.44	95.28	93.38	90.19	88.98	82.33	60.02	86.13	88.97
	aIOU	28.79	13.25	44.09	24.82	44.29	36.07	32.90	47.28	18.66	11.72	31.73	7.15
	SIM	0.501	0.363	0.546	0.399	0.728	0.654	0.684	0.715	0.356	0.347	0.597	0.545
	MAE	0.087	0.130	0.055	0.062	0.059	0.066	0.095	0.086	0.086	0.166	0.097	0.069
FRCNN [15]	**AUC**	82.55	76.01	90.30	86.72	93.05	91.51	88.13	85.02	83.29	66.35	88.18	90.74
	aIOU	28.32	15.29	33.77	24.94	39.00	33.18	31.71	45.14	17.38	16.45	33.21	10.08
	SIM	0.506	0.419	0.421	0.390	0.680	0.628	0.684	0.704	0.337	0.442	0.636	0.492
	MAE	0.093	0.129	0.083	0.070	0.073	0.075	0.097	0.091	0.099	0.145	0.091	0.072
XMF [1]	**AUC**	80.34	69.14	91.91	86.29	94.38	94.88	89.47	88.75	77.99	55.95	88.24	91.65
	aIOU	25.53	14.25	41.91	25.46	44.51	36.96	32.45	46.01	17.33	9.07	31.38	7.94
	SIM	0.489	0.382	0.512	0.411	0.729	0.689	0.688	0.710	0.340	0.307	0.623	0.576
	MAE	0.092	0.128	0.053	0.071	0.061	0.061	0.092	0.083	0.098	0.164	0.091	0.067
IAG [16]	**AUC**	74.33	82.56	84.94	81.69	92.23	93.59	91.25	92.14	75.94	76.78	90.82	78.68
	aIOU	20.30	20.23	28.70	24.61	38.37	34.67	33.91	48.05	22.48	17.32	34.20	6.06
	SIM	0.447	0.529	0.417	0.403	0.672	0.652	0.712	0.745	0.404	0.514	0.673	0.493
	MAE	0.107	0.111	0.070	0.070	0.074	0.066	0.091	0.078	0.101	0.125	0.082	0.073
LASO [7]	**AUC**	86.88	83.17	94.70	87.23	95.02	94.93	90.31	90.77	85.50	74.97	88.86	88.67
	aIOU	30.37	19.39	48.06	23.04	42.19	37.41	30.51	48.44	19.67	22.03	33.17	12.00
	SIM	0.560	0.507	0.570	0.389	0.724	0.684	0.685	0.750	0.375	0.534	0.628	0.550
	MAE	0.081	0.118	0.051	0.069	0.063	0.062	0.101	0.083	0.083	0.133	0.091	0.072
Ours	**AUC**	88.33	87.83	96.88	89.82	95.03	95.52	91.61	90.43	93.98	89.42	90.56	86.77
	aIOU	35.11	23.26	49.73	29.17	40.66	40.52	35.81	45.49	30.04	27.09	33.98	6.40
	SIM	0.635	0.558	0.578	0.470	0.727	0.717	0.730	0.734	0.544	0.684	0.647	0.554
	MAE	0.075	0.102	0.047	0.056	0.062	0.059	0.082	0.082	0.069	0.099	0.085	0.090
Method	**Metrics**	**Listen**	**Wear**	**Press**	**Cut**	**Stab**	**Carry**	**Ride**	**Clean**	**Play**	**Beat**	**Speak**	**Pull**
Baseline	**AUC**	86.16	92.00	88.45	63.82	82.10	93.82	94.71	85.87	94.46	93.35	88.27	88.48
	aIOU	17.37	53.14	22.68	12.03	22.08	46.96	30.20	35.92	22.24	45.10	39.49	40.76
	SIM	0.599	0.758	0.530	0.275	0.437	0.711	0.536	0.580	0.594	0.678	0.628	0.523
	MAE	0.089	0.044	0.095	0.133	0.081	0.055	0.024	0.088	0.049	0.042	0.088	0.020
FRCNN [15]	**AUC**	85.62	91.79	88.23	77.92	93.12	95.83	93.38	83.03	94.65	94.28	87.37	95.26
	aIOU	17.19	53.48	22.86	14.17	30.03	49.26	28.78	37.48	22.02	44.03	43.11	41.36
	SIM	0.607	0.779	0.536	0.410	0.531	0.751	0.528	0.600	0.598	0.681	0.674	0.528
	MAE	0.092	0.044	0.101	0.127	0.060	0.055	0.026	0.090	0.052	0.044	0.089	0.026
XMF [1]	**AUC**	89.88	91.88	90.33	70.67	80.85	94.41	95.17	85.59	94.79	93.52	83.01	88.75
	aIOU	21.00	51.46	23.40	16.19	24.74	49.79	30.12	35.38	22.38	45.26	39.11	36.42
	SIM	0.684	0.760	0.571	0.423	0.407	0.743	0.552	0.556	0.610	0.675	0.611	0.461
	MAE	0.076	0.044	0.092	0.116	0.081	0.053	0.025	0.096	0.052	0.041	0.096	0.028
IAG [16]	**AUC**	90.12	92.83	88.20	78.15	79.58	89.99	94.02	95.11	93.08	97.72	88.61	96.64
	aIOU	20.22	51.79	21.66	14.81	35.26	40.68	28.96	37.49	21.14	45.27	41.54	42.64
	SIM	0.668	0.767	0.532	0.504	0.503	0.657	0.534	0.644	0.574	0.708	0.635	0.603
	MAE	0.082	0.043	0.100	0.101	0.063	0.067	0.024	0.065	0.054	0.032	0.102	0.017
LASO [7]	**AUC**	89.40	92.75	88.22	91.50	84.93	98.48	96.66	82.52	94.25	96.13	86.55	98.99
	aIOU	19.58	54.16	22.08	19.04	34.72	52.77	28.25	25.26	21.70	42.70	41.16	40.28
	SIM	0.649	0.773	0.522	0.632	0.510	0.779	0.547	0.427	0.575	0.680	0.648	0.594
	MAE	0.082	0.043	0.104	0.073	0.060	0.045	0.023	0.119	0.054	0.039	0.092	0.014
Ours	**AUC**	91.90	92.00	90.96	86.42	88.32	98.84	98.60	89.79	96.04	97.86	89.18	98.69
	aIOU	22.26	53.91	26.64	16.72	36.82	52.53	45.01	37.06	25.93	46.75	44.36	36.34
	SIM	0.736	0.781	0.554	0.611	0.553	0.789	0.732	0.639	0.715	0.753	0.641	0.499
	MAE	0.065	0.045	0.080	0.084	0.064	0.037	0.018	0.063	0.038	0.031	0.077	0.019

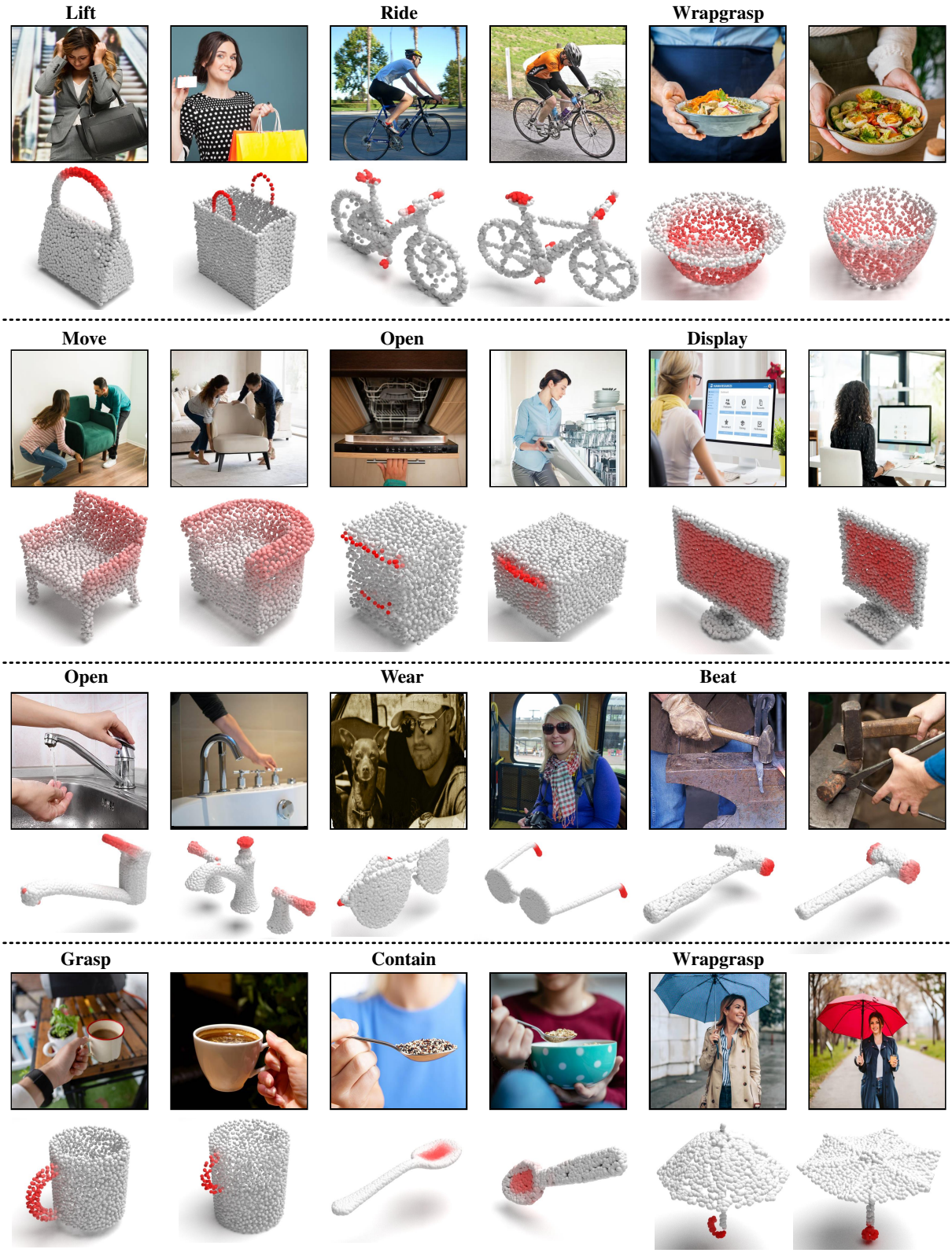

Figure 1. **More Visualization Results of GREAT for Seen Setting.**

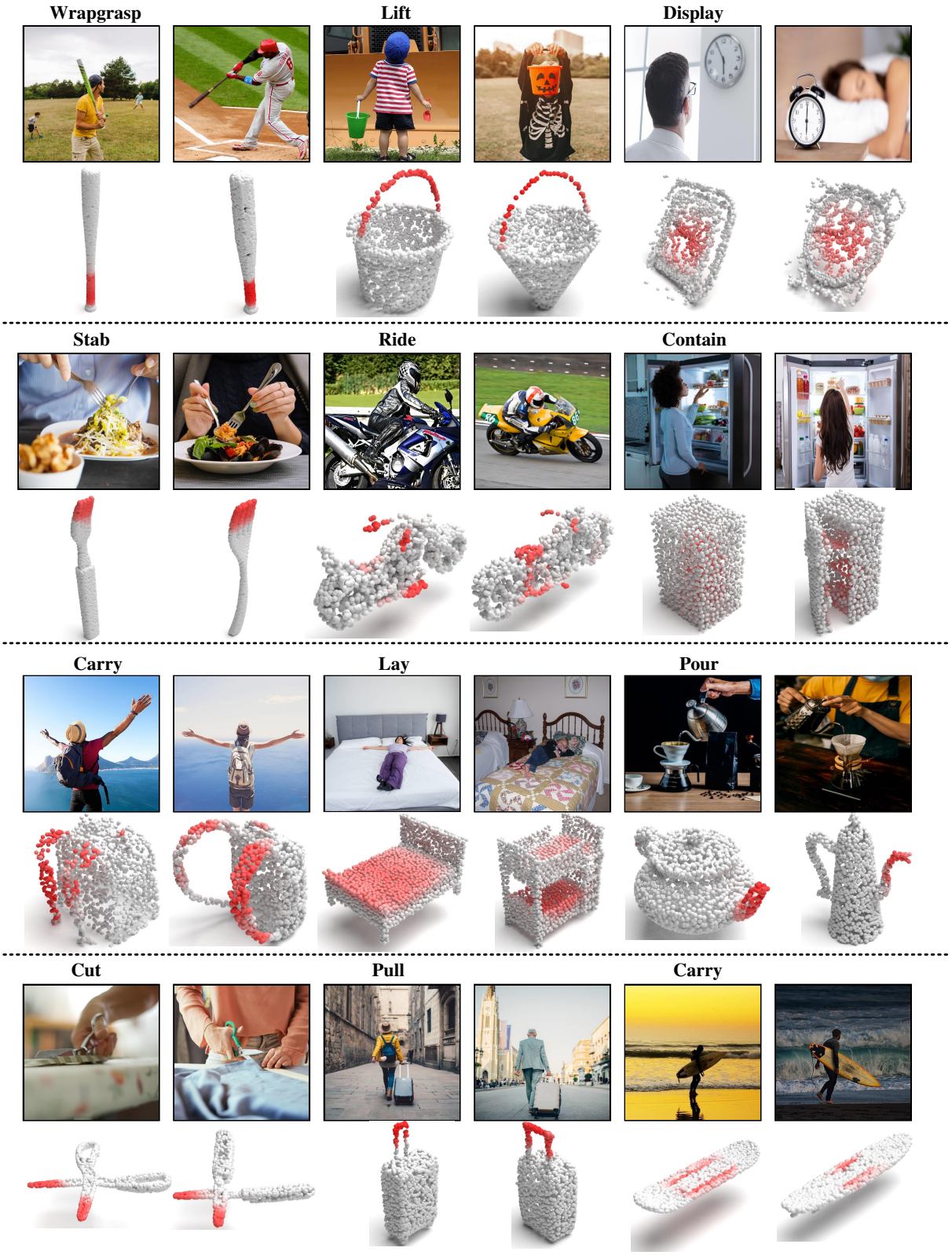

Figure 2. **More Visualization Results of GREAT for Unseen Setting.** The first four rows for **Unseen Object** setting, and the last four rows for **Unseen Affordance** setting.

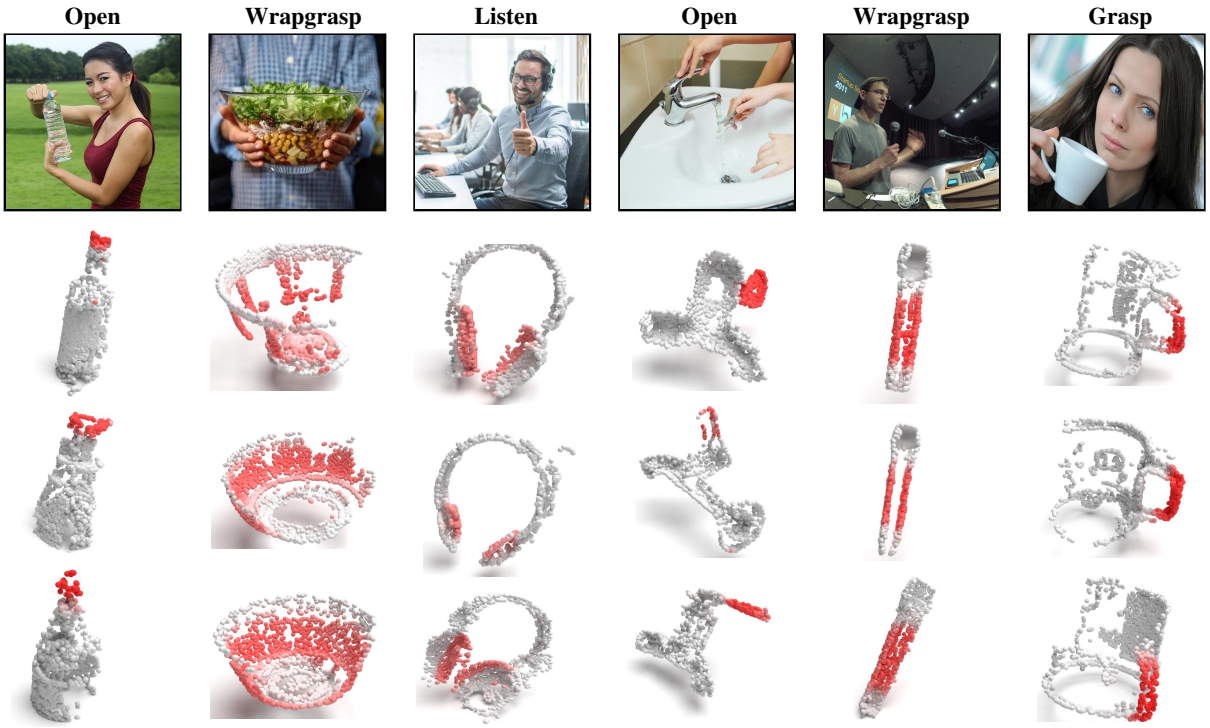

Figure 3. **Visualization Results of Partial Point Cloud.**

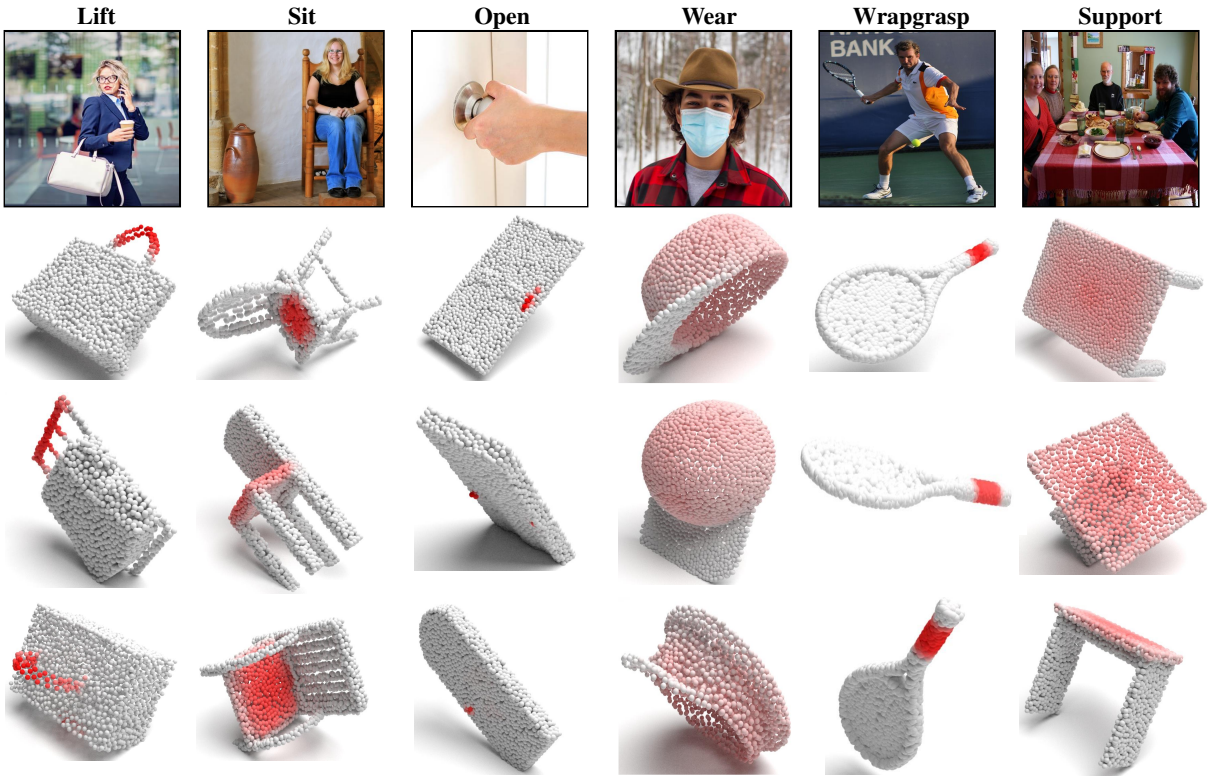

Figure 4. **Visualization Results of Rotated Point Cloud.**

References

[1] Emanuele Aiello, Diego Valsesia, and Enrico Magli. Cross-modal learning for image-guided point cloud shape completion. In *Advances in Neural Information Processing Systems*, 2022. 3, 4, 6

[2] Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*, 2024. 2

[3] Shengheng Deng, Xun Xu, Chaozheng Wu, Ke Chen, and Kui Jia. 3d affordancenet: A benchmark for visual object affordance understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021. 4

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 1

[5] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. 2

[6] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014. 2

[7] Y. Li, N. Zhao, J. Xiao, C. Feng, X. Wang, and T. Chua. Laso: Language-guided affordance segmentation on 3d object. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3, 4, 5, 6

[8] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019. 1

[9] Jorge M. Lobo, Alberto Jiménez-Valverde, and Raimundo Real. Auc: a misleading measure of the performance of predictive distribution models. *Global Ecology and Biogeography*, 17:145–151, 2008. 1

[10] Toan Nguyen, Minh Nhat Vu, An Vuong, Dzung Nguyen, Thieu Vo, Ngan Le, and Anh Nguyen. Open-vocabulary affordance detection in 3d point clouds. 2023. 5

[11] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*, 2017. 1

[12] Md.Atiqur Rahman and Yang Wang. Optimizing intersection-over-union in deep neural networks for image segmentation. In *International Symposium on Visual Computing*, 2016. 1

[13] Michael J. Swain and Dana H. Ballard. Color indexing. *International Journal of Computer Vision*, 7:11–32, 1991. 1, 2

[14] Cort J. Willmott and Kenji Matsuura. Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate Research*, 30:79–82, 2005. 1, 2

[15] Xinli Xu, Shaocong Dong, Tingfa Xu, Lihe Ding, Jie Wang, Peng Jiang, Liqiang Song, and Jianan Li. Fusionrcnn: Lidar-camera fusion for two-stage 3d object detection. *arXiv preprint arXiv:2209.10733*, 2022. 3, 4, 6

[16] Yuhang Yang, Wei Zhai, Hongchen Luo, Yang Cao, Jiebo Luo, and Zheng-Jun Zha. Grounding 3d object affordance from 2d interactions in images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10905–10915, 2023. 3, 4, 5, 6