HAMMER: Harnessing MLLM via Cross-Modal Integration for Intention-Driven 3D Affordance Grounding

Supplementary Material

Contents

Overview

This supplementary document provides extended implementation details, further experimental evaluations, details on the construction of our corrupted benchmark, and additional qualitative results for the proposed approach. The material is organized as follows. Sec. S.1 elaborates on the model architecture, training configuration, and evaluation metrics. Sec. S.2 presents supplementary experiments, which include more ablation studies, sensitivity analysis of point backbones, and a detailed assessment of robustness. The construction procedure of our corrupted benchmark is delineated in Sec. S.3. Finally, Sec. S.4 demonstrates further qualitative visualizations, including results on the PIADv2 dataset [35], predictions on corrupted point clouds, visual analysis of enhanced point features, and an examination of several failure cases.

S.1. Implementation Details

S.1.1. Model Details

Our framework employs the Qwen2.5-VL multimodal large language model (MLLM) [3] for interpreting reference images and utilizes PointNet++ [29] as the backbone for extracting geometric features from 3D objects. Specifically, we adopt the 3B parameter version of Qwen2.5-VL, which comprises 36 layers in its language module and 32 layers in the vision transformer (ViT) component. The PointNet++ backbone is configured with 3 set abstraction layers and 3 feature propagation layers to effectively capture multi-scale geometric structures from raw point clouds. The point-wise features generated by PointNet++ have a dimensionality of 512, while the contact-aware intention embedding `[CONT]` is designed with a dimension of 256. To ensure compatibility between modalities, the `[CONT]` embedding is projected via a linear layer to match the point feature dimension. All input point clouds have a fixed size of 2,048 points. The entire model is implemented within the PyTorch deep learning framework.

S.1.2. Training Details

The proposed HAMMER is trained in an end-to-end manner using the DeepSpeed [34] to accelerate the training process. To maintain numerical stability during optimization, the point cloud backbone is trained in full precision, while the remaining components of the model employ mixed precision (BF16) to reduce memory consumption and increase computational throughput. Fine-tuning of the MLLM's language component is performed using Low-Rank Adaptation (LoRA) [11] with a rank of 16 and a scaling factor of 32, whereas the vision component of the MLLM remains frozen to preserve its pre-trained representational capabilities. Training is executed on 4 NVIDIA H20 GPUs with a global batch size of 64. The model is optimized using the AdamW optimizer [24] with an initial learning rate of $1e{-}4$ and a linear learning rate scheduling strategy. The complete training procedure spans 30 epochs to ensure convergence.

S.1.3. Evaluation Metrics

To comprehensively assess the performance of intention-driven 3D affordance grounding, we employ four widely recognized evaluation metrics: average Interaction Overlap (**aIOU**), Area Under the ROC Curve (**AUC**), Similarity (**SIM**), and Mean Absolute Error (**MAE**). These metrics collectively measure different aspects of the prediction quality. The definitions of these metrics are as follows:

- **aIOU** [33]: The Interaction Overlap (IOU) is a fundamental metric for evaluating the spatial agreement between two regions at a given threshold. It is defined as the ratio of the intersection to the union of the predicted

and ground-truth areas:

$$\text{IOU} = \frac{TP}{TP + FP + FN},$$

where TP, FP, and FN represent the number of true positives, false positives, and false negatives, respectively. The aIOU metric aggregates IOU values across T different thresholds to provide a more robust evaluation, computed as:

$$\text{aIOU} = \frac{1}{T} \sum_{t=1}^{T} \text{IOU}_i.$$

This averaging reduces the sensitivity to threshold selection and offers a stable measure of overall performance.

- **AUC** [23]: The Area Under the ROC Curve (AUC) quantifies the model's ability to distinguish between positive and negative samples across all classification thresholds. The ROC curve plots the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings.
- **SIM** [37]: The Similarity (SIM) metric evaluates the distributional alignment between the predicted affordance map and the ground-truth distribution. It is computed as the sum of the minimum values at each point after normalization:

$$\texttt{SIM}(\boldsymbol{P}, \hat{\boldsymbol{P}}) = \sum_{i=1}^{N} \min(\boldsymbol{P}_i, \hat{\boldsymbol{P}}_i),$$

$$\text{where} \quad \boldsymbol{P}_i = \frac{\boldsymbol{p}_i}{\sum_{j=1}^{N} \boldsymbol{p}_j}, \quad \hat{\boldsymbol{P}}_i = \frac{\hat{\boldsymbol{p}}_i}{\sum_{j=1}^{N} \hat{\boldsymbol{p}}_j}.$$

Here, $\boldsymbol{P}$ and $\hat{\boldsymbol{P}}$ denote the normalized predicted and ground-truth affordance score distributions, respectively. SIM measures the overall congruence between the two distributions, with higher values indicating greater resemblance.

- **MAE** [42]: The Mean Absolute Error (MAE) computes the average absolute deviation between the predicted and ground-truth affordance scores. It is defined as:

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |\boldsymbol{e}_i|, \quad w.r.t. \quad \boldsymbol{e}_i = \boldsymbol{p}_i - \hat{\boldsymbol{p}}_i,$$

where N is the total number of points, and $\boldsymbol{p}_i$ and $\hat{\boldsymbol{p}}_i$ are the predicted and ground-truth affordance scores for the i-th point, respectively. MAE provides a direct measure of the average prediction error and is less sensitive to outliers compared to squared error metrics.

In summary, aIOU measures the spatial overlap of predictions and ground truth, AUC evaluates the model's capability to differentiate between positive and negative samples, SIM assesses the distributional similarity, and MAE captures the average point-wise prediction error. Together, these metrics offer a multi-faceted and holistic evaluation framework for 3D affordance grounding models .

Table S.1. **Ablation study on the hierarchical cross-modal integration mechanism.** This analysis evaluates the contribution of each integration stage to the overall performance of HAMMER.

Type	Stage I	Stage II	aIOU↑	AUC↑	SIM↑	MAE↓
Seen	✓	✓	**22.20**	**88.43**	**0.605**	**0.083**
	✗	✓	21.55	87.79	0.598	0.084
	✓	✗	21.70	88.00	0.603	0.084
	✗	✗	21.15	87.64	0.597	0.085
Unseen	✓	✓	**13.71**	**80.92**	**0.449**	0.109
	✗	✓	11.85	79.55	0.433	0.110
	✓	✗	11.99	77.99	0.442	**0.106**
	✗	✗	10.50	74.72	0.407	0.116

Table S.2. **Alternative lifting strategies.** We compare the performance of our proposed lifting strategy with two alternatives: (1) single-stage lifting, and (2) simple concatenation.

Type	Ablation	aIOU↑	AUC↑	SIM↑	MAE↓
Seen	**HAMMER**	**22.20**	**88.43**	**0.605**	**0.083**
	Single Lift	21.40	88.17	0.603	0.085
	Concat. Lift	21.05	87.95	0.597	0.084
Unseen	**HAMMER**	**13.71**	**80.92**	**0.449**	**0.109**
	Single Lift	12.26	80.64	0.441	0.111
	Concat. Lift	11.45	78.12	0.421	0.110

S.2. Additional Experiments

This section presents extended experimental analyses to comprehensively evaluate the efficacy of the proposed HAMMER. We perform ablation studies on the hierarchical cross-modal integration mechanism to quantify the contribution of each stage, investigate the sensitivity of our approach to point cloud backbones, and compare the strategies of fine-tuning versus freezing the MLLM. Furthermore, we include a comparative analysis against InteractVLM [7], report object-level and affordance-level performance on public benchmarks, and present detailed robustness performance to rigorously examine the model's stability and generalization capability

S.2.1. Ablation on Hierarchical Integration

Our hierarchical cross-modal integration mechanism comprises two sequential refinement stages. To evaluate the individual contribution of each stage, we conduct an ablation study on the PIAD benchmark [47]. As summarized in Tab. S.1, removing either stage leads to a performance degradation, which is particularly pronounced on the Unseen split. The absence of *Stage I* results in a more significant performance drop, underscoring its critical role in the overall framework. This indicates that the initial, coarse-grained enhancement is fundamental for injecting rich con-

Table S.3. **Performance analysis with different point cloud backbones.** We evaluate the capability of HAMMER using PointNet++ [29] and Uni3D [55] as 3D feature extractors.

Backbones	PIAD Seen				PIAD Unseen				PIADv2 Seen				PIADv2 Unseen Object				PIADv2 Unseen Affordance			
	aIOU↑	AUC↑	SIM↑	MAE↓	aIOU↑	AUC↑	SIM↑	MAE↓	aIOU↑	AUC↑	SIM↑	MAE↓	aIOU↑	AUC↑	SIM↑	MAE↓	aIOU↑	AUC↑	SIM↑	MAE↓
PointNet++	22.20	88.43	0.605	0.083	**13.71**	**80.92**	**0.449**	0.109	40.06	**94.19**	**0.698**	**0.063**	**24.28**	**84.78**	**0.449**	**0.112**	13.28	**72.21**	**0.302**	**0.128**
Uni3D	**24.67**	**89.77**	**0.635**	**0.078**	12.28	77.75	0.412	**0.102**	**40.66**	93.94	0.690	0.065	24.02	81.07	0.426	0.131	**14.49**	71.00	0.272	**0.128**

textual cues from the reference image into the point cloud features. The subsequent *Stage II*, which performs a fine-grained refinement, provides complementary object-level semantic information, further boosting the performance. The initial stage focuses on integrating broader contextual information, while the subsequent refinement enables a more detailed, semantic-aware adjustment, resulting in a more accurate cross-modal object representation for effective affordance grounding.

S.2.2. Alternative Lifting Strategies

To validate the effectiveness of our multi-granular geometry lifting, we compare it against two alternative strategies: (1) *Single Lift*, which directly lifts the intention embedding with the last stage of point features, and (2) *Concat. Lift*, which simply concatenates the embedding with point features. As shown in Tab. S.2, our proposed multi-granular lifting consistently outperforms both alternatives across all evaluation metrics on the PIAD benchmark. The single-stage lifting approach may fail to capture the complementary information present at different levels of the point feature hierarchy, while the concatenation method may not effectively leverage the spatial characteristics 3D features.

S.2.3. Sensitivity to Point Backbones

In this section, we analyze the performance of our HAMMER with different 3D point cloud backbones. We evaluate two widely used architectures: PointNet++ [29] and Uni3D [55]. Specifically, the PointNet++ is trained from scratch, while the Uni3D is initialized with a weight pre-trained on large-scale 3D datasets, following the practice of SeqAfford [52]. The results are summarized in Tab. S.3. On the PIAD dataset, Uni3D achieves higher performance on the Seen split, but PointNet++ yields superior results on the Unseen split. This discrepancy may stem from Uni3D's pre-trained weights potentially leading to overfitting on the seen categories, whereas training PointNet++ from scratch may facilitate better adaptation to the specific characteristics of the PIAD benchmark. On the PIADv2 dataset, which contains a broader variety of object instances, PointNet++ consistently outperforms Uni3D across all three splits in most metrics, underscoring its stronger generalization capability in handling diverse 3D objects and affordance types.

Table S.4. **Comparison of MLLM fine-tuning strategies.** We evaluate the performance when applying LoRA fine-tuning to the MLLM versus keeping its weights entirely frozen during training.

Type	Ablation	aIOU↑	AUC↑	SIM↑	MAE↓
Seen	***w/ Fine-tuning***	**22.20**	**88.43**	**0.605**	**0.083**
	w/o Fine-tuning	19.67	85.53	0.568	0.092
Unseen	***w/ Fine-tuning***	**13.71**	**80.92**	**0.449**	**0.109**
	w/o Fine-tuning	7.55	64.89	0.311	0.142

Table S.5. **Comparison to InteractVLM [7] on PIAD.** We evaluate the performance of our HAMMER against InteractVLM across both Seen and Unseen splits of the PIAD dataset.

Methods	Size	Seen				Unseen			
		aIOU↑	AUC↑	SIM↑	MAE↓	aIOU↑	AUC↑	SIM↑	MAE↓
InteractVLM	13B	21.20	86.47	**0.627**	**0.081**	8.50	75.45	0.414	**0.099**
HAMMER	3B	**22.20**	**88.43**	0.605	0.083	**13.71**	**80.92**	**0.449**	0.109

S.2.4. Fine-tune *vs.* Freeze MLLM

In the training of HAMMER, the language component of the MLLM is fine-tuned using LoRA, while the vision component remains frozen. To evaluate its efficacy, we conduct a comparative analysis between fine-tuning the MLLM's language component and freezing the entire MLLM. The experimental results, detailed in Tab. S.4, clearly demonstrate that fine-tuning yields a significant performance improvement across all evaluation metrics on both the Seen and Unseen splits. The performance gain is particularly substantial on the Unseen split, which is critical for assessing model generalization. Specifically, fine-tuning leads to a remarkable increase in aIOU from 7.55 to 13.71, AUC from 64.89 to 80.92, and SIM from 0.311 to 0.449, accompanied by a reduction in MAE from 0.142 to 0.109. This pronounced improvement underscores the necessity of adapting the MLLM to the specific task of intention-driven 3D affordance grounding. By fine-tuning the language component with LoRA, the model learns to better interpret and correlate the linguistic descriptions of interactions with the visual cues from the image and the geometric structure of the 3D point cloud. This enhanced cross-modal understand-

Table S.6. **Evaluation Metrics in PIAD Seen.** Results of each affordance type in the Seen. *cont.*, *supp.*, *wrap.*, and *disp.* denote *contain*, *support*, *wrapgrasp*, and *display*, respectively.

Method	Metrics	grasp	cont.	lift	open	lay	sit	supp.	wrap.	pour	move	disp.	push	listen	wear	press	cut	stab
XMFNet	**aIOU↑**	12.82	10.41	13.24	9.17	17.00	26.65	7.89	5.24	7.81	6.46	17.25	2.68	5.31	5.34	7.87	6.35	5.88
	AUC↑	62.98	79.97	78.93	79.74	88.93	94.91	84.56	56.47	82.15	54.97	82.91	63.74	74.93	70.57	88.18	79.17	69.97
	SIM↑	0.415	0.401	0.334	0.184	0.427	0.619	0.659	0.569	0.399	0.391	0.508	0.535	0.433	0.565	0.237	0.427	0.394
	MAE↓	0.121	0.131	0.138	0.131	0.129	0.093	0.119	0.128	0.159	0.192	0.127	0.083	0.164	0.129	0.113	0.135	0.152
IAGNet	**aIOU↑**	16.83	17.12	31.95	28.39	31.80	37.72	12.04	6.02	20.33	5.57	30.57	1.79	15.59	6.55	14.42	12.95	9.48
	AUC↑	77.53	83.84	95.05	90.89	93.54	95.94	84.58	66.71	86.02	63.09	89.29	84.71	87.13	71.16	89.46	86.69	76.4
	SIM↑	0.530	0.534	0.368	0.401	0.685	0.723	0.716	0.571	0.525	0.443	0.657	0.418	0.671	0.563	0.402	0.507	0.280
	MAE↓	0.108	0.093	0.030	0.044	0.081	0.066	0.100	0.143	0.096	0.174	0.084	0.085	0.090	0.129	0.059	0.085	0.099
GREAT	**aIOU↑**	12.99	14.24	40.99	19.68	28.65	39.37	11.34	5.09	7.95	4.90	26.30	3.32	14.73	6.42	12.99	12.34	6.31
	AUC↑	76.65	79.96	94.09	89.02	94.03	96.09	85.51	68.00	89.47	67.70	88.47	88.08	88.02	74.29	88.50	88.08	79.10
	SIM↑	0.447	0.513	0.450	0.342	0.719	0.722	0.725	0.610	0.372	0.534	0.618	0.474	0.656	0.575	0.378	0.495	0.370
	MAE↓	0.126	0.109	0.030	0.067	0.083	0.069	0.105	0.136	0.132	0.162	0.096	0.089	0.097	0.142	0.070	0.089	0.104
HAMMER	**aIOU↑**	21.65	22.00	36.71	24.93	29.57	38.86	11.85	5.69	23.02	9.98	30.60	3.85	13.41	5.08	13.15	15.13	37.87
	AUC↑	85.63	89.04	80.57	92.14	92.81	96.47	85.36	69.01	94.93	79.46	91.10	87.60	87.81	72.26	89.49	93.59	99.87
	SIM↑	0.656	0.606	0.367	0.392	0.651	0.718	0.720	0.661	0.587	0.606	0.674	0.591	0.642	0.568	0.399	0.688	0.574
	MAE↓	0.090	0.086	0.078	0.054	0.091	0.066	0.100	0.128	0.083	0.134	0.087	0.083	0.096	0.146	0.061	0.061	0.018

ing enables the MLLM to capture the intricate relationships between language, image, and 3D geometry more effectively, which in turn translates to superior affordance prediction capabilities, especially when generalizing to novel objects and scenarios not encountered during training.

S.2.5. Comparison with InteractVLM

In Tab. S.5, we present a comparison between our HAMMER and InteractVLM [7] on the PIAD benchmark. It is noteworthy that InteractVLM employs a 13B parameter version of LISA [16] as its MLLM backbone to enhance performance. Despite utilizing a significantly smaller 3B parameter MLLM, our HAMMER consistently surpasses InteractVLM across both the Seen and Unseen splits in terms of aIOU and AUC metrics. Specifically, on the Seen split, HAMMER achieves an aIOU of 22.20 and an AUC of 88.43, outperforming InteractVLM, which yields 21.20 and 86.47, respectively. The performance advantage is more pronounced on the challenging Unseen split, where HAMMER attains an aIOU of 13.71 and an AUC of 80.92, substantially exceeding InteractVLM's 8.50 and 75.45. These results underscore the efficacy of our hierarchical cross-modal integration and geometry lifting modules in enhancing 3D affordance grounding capabilities, even with a more compact MLLM architecture.

S.2.6. Detailed Performance on Public Datasets

In the main paper, we report the overall performance of different methods on PIAD [47] and PIADv2 [35]. In this section, we provide a fine-grained analysis by breaking down the results according to object categories and affordance types, offering deeper insights into the specific strengths and generalization capabilities of HAMMER relative to existing approaches.

Tab. S.6 presents the per-affordance performance on the PIAD Seen split. Our method achieves superior results across most affordance categories, demonstrating its consistent effectiveness in accurately grounding diverse functional properties. The analysis on the Unseen split in Tab. S.7 reveals that HAMMER achieves significant improvements over baseline methods, with particularly notable gains on challenging affordance types such as *contain*, *wrapgrasp*, and *display*. This underscores the model's strong generalization capacity to novel affordances beyond the training distribution. Further evaluation on the more challenging PIADv2 benchmark corroborates these findings. As shown in Tab. S.8, which details performance per affordance type on the Unseen Affordance split, HAMMER attains leading performance across the majority of categories. This indicates its ability to effectively capture the nuanced characteristics of various affordances, even when they are completely unseen during training. Similarly, the per-category results on the Unseen Object split (Tab. S.10) highlight its adaptability to diverse 3D shapes and structural configurations. Collectively, these results provide comprehensive evidence validating the efficacy of HAMMER in intention-driven 3D affordance grounding. The consistent performance gains across different datasets and splits confirm the generalizability of our proposed framework in addressing various challenges in 3D affordance perception.

S.2.7. Detailed Robustness Performance

In the main manuscript, we present a comparative analysis of the overall robustness between our method HAMMER and the baseline GREAT [35] on our newly established cor-

Table S.7. **Evaluation Metrics in PIAD Unseen.** Results of each affordance type in the Unseen. *cont.*, *wrap.*, and *disp.* denote *contain*, *wrapgrasp*, and *display*, respectively.

Method	Metrics	cont.	lay	lift	wrap.	open	disp.	stab	grasp	press	cut
XMFNet	**aIOU**↑	6.29	15.10	7.29	1.42	4.32	6.20	6.12	3.97	5.71	13.95
	AUC↑	67.98	84.02	68.45	45.74	78.53	62.20	76.92	59.19	69.32	85.87
	SIM↑	0.412	0.503	0.403	0.451	0.156	0.075	0.351	0.278	0.270	0.435
	MAE↓	0.137	0.135	0.144	0.156	0.094	0.240	0.087	0.117	0.124	0.078
IAGNet	**aIOU**↑	7.24	18.12	8.47	1.89	12.28	16.28	10.39	4.79	4.22	21.47
	AUC↑	67.96	84.82	71.10	56.39	90.91	85.51	98.83	78.60	68.07	95.95
	SIM↑	0.430	0.525	0.407	0.556	0.227	0.393	0.437	0.533	0.194	0.599
	MAE↓	0.125	0.130	0.143	0.150	0.050	0.130	0.044	0.102	0.122	0.057
GREAT	**aIOU**↑	12.08	20.57	9.40	1.24	2.23	1.36	0.07	0.23	12.85	0.36
	AUC↑	80.07	87.09	74.12	39.03	75.61	19.97	49.58	51.13	86.68	49.23
	SIM↑	0.444	0.533	0.410	0.326	0.102	0.054	0.056	0.201	0.417	0.101
	MAE↓	0.112	0.115	0.124	0.176	0.069	0.211	0.068	0.118	0.098	0.079
HAMMER	**aIOU**↑	17.57	23.62	11.96	2.48	15.00	22.28	3.36	0.61	14.55	3.37
	AUC↑	86.08	84.60	73.68	54.47	91.53	91.09	96.06	33.90	90.98	79.51
	SIM↑	0.521	0.530	0.394	0.553	0.275	0.462	0.212	0.413	0.430	0.253
	MAE↓	0.106	0.124	0.139	0.163	0.069	0.111	0.154	0.179	0.065	0.173

Table S.8. **Evaluation Metrics in PIADv2 Unseen Affordance.** Results of each affordance type in the Unseen Affordance split.

Method	Metrics	Carry	Listen	Lay	Pour	Cut	Pull
XMFNet	**aIOU**↑	5.89	3.89	10.93	6.85	5.77	24.52
	AUC↑	54.08	56.07	73.16	63.97	44.85	91.40
	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
IAGNet	**aIOU**↑	8.10	3.71	10.47	4.92	4.40	38.27
	AUC↑	63.98	54.13	69.94	59.89	49.97	93.72
	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
GREAT	**aIOU**↑	12.59	2.48	10.66	11.28	8.53	41.53
	AUC↑	82.13	51.36	77.53	72.82	52.21	97.39
	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
HAMMER	**aIOU**↑	17.53	1.99	10.42	10.06	16.35	41.56
	AUC↑	75.33	38.11	75.85	81.31	87.95	96.67
	SIM↑	0.378	0.072	0.393	0.194	0.262	0.537
	MAE↓	0.141	0.229	0.132	0.105	0.090	0.025

Table S.9. **Statistics of the constructed corrupted benchmark.**

#	Object Category	Affordance Type	Num
1	TrashCan	contain, open, pour	69
2	Door	open, push	47
3	Display	display	52
4	Earphone	grasp, listen	70
5	Vase	contain, wrapgrasp	90
6	Hat	grasp, wear	66
7	Bottle	contain, wrapgrasp, pour, open	225
8	Keyboard	press	25
9	Knife	grasp, cut, stab	138
10	Refrigerator	contain, open	53
11	Laptop	display, press	112
12	Dishwasher	contain, open	39
13	Bowl	contain, wrapgrasp, pour	83
14	Clock	display	9
15	StorageFurniture	contain, open	92
16	Scissors	grasp, cut, stab	29
17	Table	support, move	194
18	Bag	grasp, contain, lift, open	50
19	Faucet	grasp, open	95
20	Mug	grasp, contain, wrapgrasp, pour	126
21	Chair	sit, move	392
22	Bed	lay, sit	56
23	Microwave	contain, open	47
Total	**23 Categories**	**17 Affordance Types**	**2159**

rupted benchmark. This section provides a detailed evaluation by examining the performance under individual corruption types, including scale, jitter, rotation, local dropout, global dropout, local additive noise, and global additive noise. Each corruption type is assessed across five severity levels, ranging from 0 to 4. This benchmark serves as a proxy for real-world imperfections in 3D data, such as sensor noise, occlusions, and partial observations. As shown in Tab. S.11, HAMMER consistently outperforms GREAT under all corruption conditions and severity levels. This robust performance indicates the effectiveness of our approach in handling diverse corruptions in 3D data, demon-

Table S.10. **Evaluation Metrics in PIADv2 Unseen Object.** Results of each object category in the Unseen Object split. *Base.*, *Motor.*, *Refri.*, Scus. and *Skat.* denote Baseballbat, Motorcycle, Refrigerator, Scissors, and Skateboard, respectively.

Method	Metrics	Base.	Bucket	Clock	Fork	Kettle	Laptop	Mop	Motor.	Refri.	Scis.	Skat.
XMFNet	**aIOU**↑	37.62	10.26	19.13	27.23	5.50	8.66	20.53	5.54	9.41	10.24	18.38
	AUC↑	76.75	59.73	77.86	81.93	61.74	73.52	79.09	69.02	84.22	56.31	81.80
	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
IAGNet	**aIOU**↑	42.84	12.56	13.48	22.60	2.65	6.95	37.66	9.30	14.43	7.37	4.61
	AUC↑	85.32	65.52	69.21	73.63	54.38	78.17	92.09	79.74	80.63	56.62	58.84
	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
GREAT	**aIOU**↑	41.93	28.75	20.23	31.83	7.33	5.17	32.37	17.79	13.45	12.97	8.72
	AUC↑	82.73	83.41	84.00	89.28	79.33	74.98	91.80	95.73	78.69	64.83	59.50
	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
HAMMER	**aIOU**↑	49.99	36.68	18.68	40.51	6.93	13.27	37.62	9.41	21.84	13.99	17.43
	AUC↑	87.17	91.49	80.55	87.50	59.38	87.60	93.39	83.17	92.15	63.48	91.65
	SIM↑	0.673	0.525	0.434	0.530	0.126	0.383	0.653	0.230	0.440	0.307	0.512
	MAE↓	0.133	0.062	0.150	0.118	0.140	0.084	0.084	0.037	0.069	0.181	0.158

strating its potential for real-world applications where incomplete or noisy point clouds are common.

S.3. Corrupted Benchmark Construction

To evaluate the robustness of language-prompted 3D affordance grounding models, GEAL [25] introduced a set of corruption types applied to 3D point clouds. However, our work addresses intention-driven 3D affordance grounding, which requires both 3D point clouds and their corresponding reference images. Accordingly, we construct a corrupted benchmark by applying the same corruption types and severity levels as defined in GEAL to the 3D point clouds. The corruption types include scale, jitter, rotation, dropout-local, dropout-global, add-local, and add-global. Each type is evaluated under five severity levels, ranging from 0 to 4. For implementation details regarding the operation of these corruptions, we refer readers to the supplementary material of GEAL [25]. Since the PIAD dataset provides multiple interaction images for each object, we randomly select one reference image from the available images for each corrupted 3D object. For every combination of corruption type and severity level, the benchmark contains 2,159 image-point cloud pairs, covering 23 object categories and 17 affordance types. The detailed statistics of the corrupted benchmark are summarized in Tab. S.9.

S.4. More Visualizations

S.4.1. Visualization on PIADv2

To further illustrate the efficacy of our HAMMER, we provide more qualitative comparisons on PIADv2 [35], complementing the results presented in the main manuscript. Fig. S.2 showcases additional visual examples on the Seen split of PIADv2, comparing our method with GREAT [35]. The results indicate that HAMMER consistently generates more accurate and complete affordance maps, which align more closely with the ground-truth annotations. For instance, in the case of the broom, our model correctly identifies the handle region suitable for the *wrapgrasp* affordance, whereas GREAT erroneously highlights the cleaning area. Similarly, for the hammer, our approach precisely localizes the beating surface, while GREAT produces a less accurate region that includes portions of the handle. Furthermore, Fig. S.3 provides more qualitative results on the Unseen Object and Unseen Affordance splits of PIADv2. These examples show the strong generalization capability of HAMMER when confronted with novel object categories and affordance types. For example, when presented with a bucket (an unseen object), our model accurately identifies the handle region for the *lift* affordance, but GREAT yields an incomplete affordance map that omits critical parts. In another case involving the novel affordance *listening* applied to earphones, our method successfully highlights the

Table S.11. **Detailed robustness performance under different levels of corruptions.** We evaluate the robustness of our HAMMER and GREAT [35] on 7 corruption types. Each corruption type is evaluated under 5 severity levels (0-4). The results demonstrate that HAMMER consistently outperforms GREAT across all corruption types and severity levels, highlighting its superior robustness in handling corrupted 3D objects.

Corrupt Type	Level	GREAT				HAMMER			
		aIOU↑	AUC↑	SIM↑	MAE↓	aIOU↑	AUC↑	SIM↑	MAE↓
Scale	0	13.85	78.63	0.497	0.115	19.76	86.04	0.604	0.091
	1	13.59	77.89	0.495	0.116	19.26	85.75	0.598	0.093
	2	13.50	77.82	0.491	0.116	19.15	85.83	0.594	0.094
	3	13.11	77.21	0.486	0.117	18.94	85.52	0.591	0.095
	4	13.01	77.30	0.483	0.117	18.81	85.51	0.591	0.094
Jitter	0	14.51	79.79	0.510	0.114	20.13	86.57	0.609	0.090
	1	13.30	78.66	0.492	0.113	18.81	85.79	0.588	0.092
	2	11.69	75.99	0.463	0.114	17.35	84.61	0.566	0.095
	3	10.13	73.39	0.433	0.115	15.86	83.61	0.547	0.098
	4	8.74	70.42	0.407	0.117	14.64	82.25	0.531	0.101
Rotate	0	14.62	80.07	0.512	0.114	20.30	86.74	0.613	0.089
	1	14.00	79.17	0.505	0.114	19.56	86.33	0.634	0.091
	2	12.88	77.43	0.484	0.116	18.55	85.65	0.588	0.094
	3	11.55	75.34	0.467	0.117	17.46	84.54	0.571	0.097
	4	9.98	72.94	0.445	0.119	15.77	83.30	0.549	0.102
Dropout-Local	0	9.24	73.50	0.435	0.123	18.08	84.67	0.568	0.098
	1	8.83	73.07	0.426	0.122	18.29	84.81	0.570	0.098
	2	8.92	72.87	0.423	0.122	18.07	84.78	0.571	0.098
	3	8.38	71.73	0.416	0.122	17.90	84.54	0.568	0.098
	4	7.97	70.74	0.402	0.121	17.54	84.16	0.561	0.099
Dropout-Global	0	14.72	79.86	0.513	0.113	20.47	86.88	0.616	0.089
	1	14.49	79.63	0.509	0.112	20.52	86.94	0.616	0.089
	2	14.13	79.08	0.504	0.112	20.38	86.84	0.613	0.089
	3	13.40	77.73	0.488	0.112	20.40	86.66	0.611	0.089
	4	11.38	75.00	0.456	0.113	20.14	86.47	0.601	0.091
Add-Local	0	12.53	76.17	0.466	0.110	18.98	86.35	0.584	0.091
	1	11.68	74.95	0.445	0.110	18.26	86.32	0.568	0.092
	2	11.41	74.97	0.435	0.109	17.68	86.03	0.553	0.093
	3	11.00	74.67	0.424	0.108	16.97	85.81	0.536	0.094
	4	10.87	75.03	0.419	0.107	16.46	85.67	0.521	0.096
Add-Global	0	12.04	75.19	0.469	0.108	19.90	86.65	0.597	0.089
	1	10.35	72.39	0.442	0.108	19.46	86.57	0.588	0.089
	2	9.05	70.66	0.425	0.108	18.98	86.20	0.576	0.090
	3	8.63	69.35	0.416	0.108	18.59	86.28	0.567	0.091
	4	8.06	68.65	0.406	0.108	18.53	86.09	0.561	0.091

appropriate component, while GREAT's prediction includes irrelevant regions. These visual comparisons reinforce the effectiveness of HAMMER.

S.4.2. Visualization of Point Features

In the main paper, we analyze the influence of our hierarchical cross-modal integration mechanism on point features by visualizing two examples of PCA-reduced features. This section provides additional visualizations in Fig. S.4 to offer a more comprehensive evaluation. The results clearly demonstrate that the proposed integration mechanism successfully incorporates contextual information from the reference image into the point features, yielding more discriminative and semantically enriched representations. For the bed shown in the second row, the point features are enhanced through our hierarchical integration form distinct clusters corresponding to key components such as the bed frame and other parts, which are essential for accurately grounding the *lay* affordance. In contrast, the original features without integration appear more dispersed and less

structured. Similarly, for the laptop and the bottle depicted in the first row, the integrated features exhibit improved separation between different functional regions, such as the keyboard and screen of the laptop, and the body and cap of the bottle. This enhancement in feature representation contributes to more precise affordance predictions, as supported by the qualitative results presented in the main text.

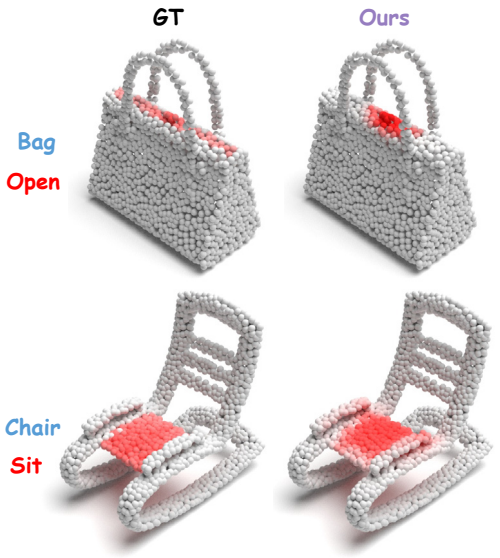

Figure S.1. **Failure cases of our HAMMER model on PIAD.**

S.4.3. Visualization of Corrupted Point Clouds

Complementing the example in the main paper, we provide further qualitative results in Fig. S.5 to illustrate the corrupted point clouds utilized for robustness assessment. In the case of the *sit* affordance on the chair, our HAMMER accurately identifies the seat region despite the presence of jitter and dropout corruptions, whereas GREAT [35] yields incomplete affordance maps. For the *grasp* affordance on the earphone, HAMMER effectively localizes the handle area under rotation and additive noise corruptions, while GREAT's predictions are less precise and incorporate irrelevant regions. These additional visualizations show the robustness of HAMMER in handling diverse corruptions in 3D point clouds, enabling reliable affordance grounding in challenging scenarios.

S.4.4. Failure Cases

Although the proposed framework achieves competitive performance, it still exhibits certain limitations in challenging scenarios. Fig. S.1 illustrates two representative failure cases. In the first example, which involves the *open* affordance on a bag, the model fails to accurately localize the full zipper region suitable for opening and only activates in a limited area. In the second example, regarding the *sit* affordance on a chair, the model incorrectly predicts the handrest region as part of the sittable area, indicating a misalignment between the inferred interaction and the object geometry. The error seems to arise from ambiguity in the intention embedding derived from the image, where the depicted interaction introduces uncertainty in distinguishing between relevant and irrelevant parts. These cases highlight specific challenges in disambiguating subtle visual cues and refining spatial awareness in complex interaction scenarios. Future work may explore more robust intention representation learning and enhanced geometric reasoning to overcome these limitations.

Figure S.2. **Visualization on PIADv2 Seen.** We compare our HAMMER with GREAT [35] on several examples from the Seen subset of PIADv2.

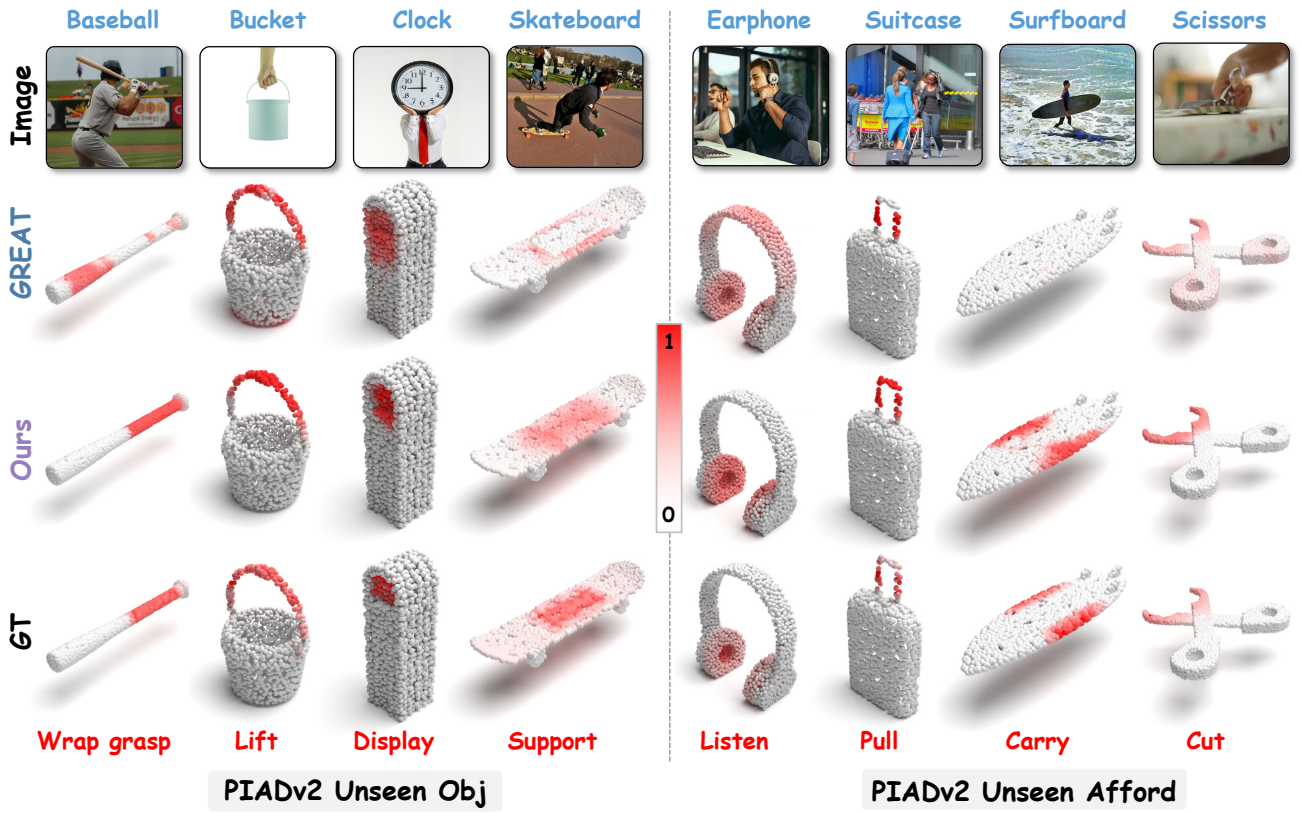

Figure S.3. **Visualization on PIADv2 Unseen Object and Unseen Affordance subsets.** We compare our HAMMER with GREAT [35] on several examples from the Unseen Object and Unseen Affordance subsets of PIADv2.

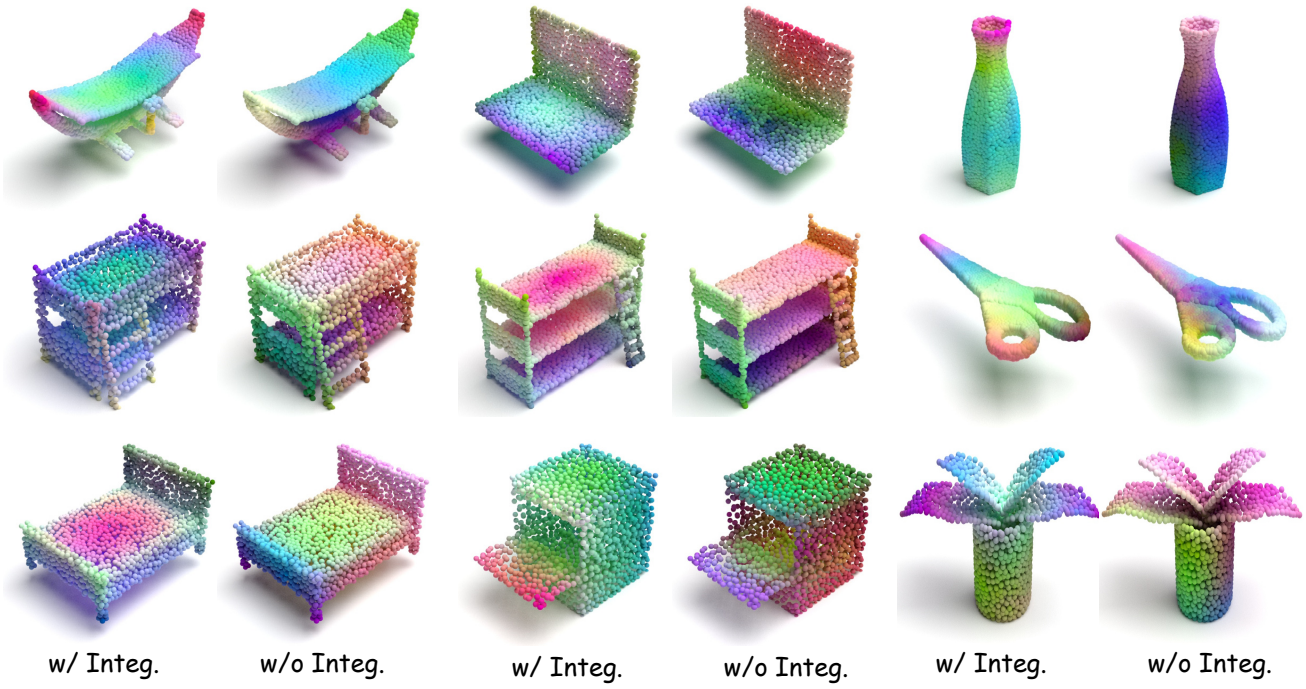

Figure S.4. **Visualization of point features with and without hierarchical cross-modal integration.** We visualize the PCA-reduced point features for several examples to demonstrate the effectiveness of our proposed hierarchical cross-modal integration mechanism.

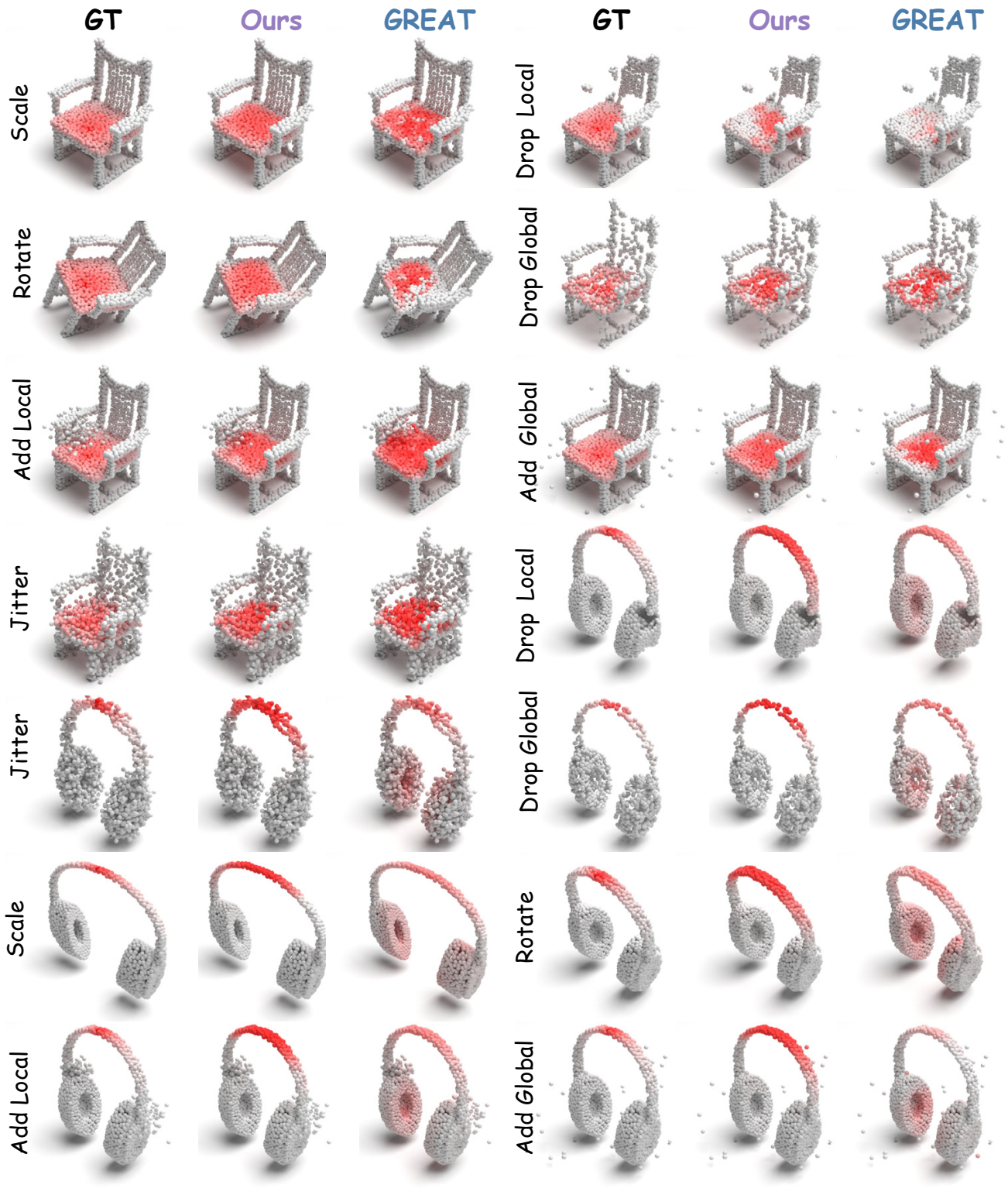

Figure S.5. **Visualization of corrupted point clouds.** We illustrate two additional examples of corrupted point clouds from our constructed benchmark.