Appendix

In this section, we present more details on the implementation and discuss additional experimental results. The following is an overview of the organization of the appendix.

A1. Implementation Details

We discussed most of the training and implementation details in the main paper. In this section, we summarize all hyperparameters in Table A1. Some parameters are shared across all training stages, while others are stage-specific. The values of these hyperparameters are chosen based on a rigorous sensitivity analysis, while some are derived from well-established literature.

Next, we present the prompt used for GPT-4 evaluation in Prompt 1. This prompt is similar to what existing methods have used for this evaluation.

Table A1. List of model-specific hyperparameters.

Stage	Hyperparam.	Value
	β	0.25
	λ	0.50
	Codebook size	8192
	Warm-up ratio	0.03
	Weight decay	0.05
	Number of sampled points, N_s	512
	Number of neighbours, K_g	81
1	Epoch	3
	Lr. rate	4×10^{-4}
	Batch size	128
2	Epoch	3
	batch size	32
	Lr. rate	2×10^{-5}
3	α	0.95
	Epoch	1
	batch size	8
	Lr. rate	1×10^{-6}
	number of generated responses, m	8

Prompt 1:

```
You are a helpful AI assistant.
Now, I will give you a question, its type, an
answer from the model, and an answer from the
label.
Your task is to focus only on these two answers
and determine whether they convey the same
information for the given type of question.
Your response should be a single confidence score
ranging from 0 to 100.
This score evaluates how closely the two answers
describe the same thing.
Follow the scoring standard demonstrated below.
Here are several example question-answer pairs
with their confidence scores:

Question 1: How many oranges will there be if
1/3 of them are removed?
Question type: Knowledge
Answer from model: There will be 6 left.
Answer from label: As there are 9 oranges in
total, there will be 6 oranges left if 1/3 of
them are removed.
Confidence score: 100

Question 2: What is this object?
Question type: General Visual Recognition
Answer from model: This is a bathtub.
Answer from label: This is a dirty bathtub.
Confidence score: 80

Question 3: What is this object?
Question type: General Visual Recognition
Answer from model: This is a bottle of water.
Answer from label: This is a bottle of oil.
Confidence score: 50

Question 4: What is the boy holding in his right
hand?
Question type: Spatial Recognition
Answer from model: He is holding a white cup in
his right hand.
Answer from label: He is holding a sword in his
right hand.
Confidence score: 0

Next, you will be given:
Question: {},
Question type: {},
Answer from model: {},
Answer from label: {}.
Output only the confidence score as a number,
without any words.
```

Table A2. **Performance comparison with existing methods.** We adopt the following table from [17] and follow the notations and categorization defined by them. Here, "Specialist Model" refers to models specifically designed for individual tasks such as 3D question answering, 3D dense captioning, or referring segmentation. "Finetuned 3D MLLMs" denotes models that are jointly trained and subsequently fine-tuned on each dataset before evaluation. "3D MLLMs" represents models trained on multiple tasks without task-specific fine-tuning. "PC" stands for point cloud, and "I" denotes multi-view images. Note that the results of LEO [28] on ScanQA are shown in gray and excluded from direct comparison, as the model uses a different setting that accesses ground-truth objects related to the questions. All models use the 7B parameter configuration.

Method	Modality	ScanQA (val)			Scan2Cap (val)		
		B-4↑	M↑	R↑	B-4↑	M↑	R↑
Specialist Models							
ScanQA [1]	PC	10.1	13.1	33.3	-	-	-
3D-VLP [33]	PC	11.2	13.5	34.5	32.3	24.8	51.5
3D-VisTA [81]	PC	10.4	13.9	**45.7**	34.1	26.8	55.0
Scan2Cap [11]	PC	-	-	-	23.3	22.0	44.8
MORE [32]	PC	-	-	-	22.9	21.7	44.4
SpaCap3D [63]	PC	-	-	-	25.3	22.3	45.4
D3Net [7]	PC	-	-	-	30.3	24.4	51.7
UniT3D [12]	PC	-	-	-	27.2	21.9	46.0
3DJCG [4]	PC	-	-	-	31.0	24.2	50.8
Vote2Cap-DETR [8]	PC	-	-	-	34.5	26.2	54.4
Finetuned 3D MLLMs							
3D-LLM [25]	PC+I	12.0	14.5	35.7	-	-	-
Scene-LLM [21]	PC+I	12.0	16.8	40.0	-	-	-
LL3DA [23]	PC	13.5	15.9	37.3	36.8	26.0	55.1
3D MLLMs							
LEO [28]	PC+I	13.2	20.0	49.2	**38.2**	27.9	58.1
Scene-LLM [21]	PC+I	11.7	15.8	35.9	-	-	-
Chat-Scene [27]	PC+I	14.3	18.0	41.6	36.4	28.0	58.1
Grounded 3D-LLM [10]	PC	13.4	-	-	35.5	-	-
3D-LLaVA [78]	PC	17.1	18.4	43.1	36.9	27.1	57.7
SAGE-7B	PC	**17.5**	**19.6**	44.5	37.8	**28.1**	**58.6**

A2. Discussion

A2.1. Non-comparable Existing Method

While we discuss many existing methods in the related work, we do not directly compare our approach with all of them in the main results. This is because some of the existing methods differ substantially in terms of both their learning strategies and the types of data they use. For instance, some methods are first trained on dense prediction tasks [17] using segmentation data. Another major source of disparity lies in the training data. While our method, and a few others [68] adhere strictly to single training dataset [16], some approaches [48] combine multiple datasets from different sources [6, 14, 16, 20, 60, 65] to increase data volume, and others propose data generation pipelines to dynamically curate additional samples [34]. Additionally, prior works [22, 40, 47, 69, 70, 73, 75, 77, 80] in this literature focused on 3D encoders with any multimodal capabilities utilizing well-established training pipelines [30, 52]. In the main table of the paper, we only compare against methods that follow similar datasets and training protocols [68] as ours. Nonetheless, we provide additional results and further discussion on these disparities in the appendix.

Table A3. Total training time.

Model	Training time (h)
PointLLM	26.1
SAGE*	18.0
SAGE	27.4

A3. Additional Results

In this section, we present additional experimental results for our proposed model. We first compare SAGE's performance with a broader set of state-of-the-art methods on 3D captioning and 3D VQA tasks across two datasets. We then report multi-run results on the Objaverse 3D cap-

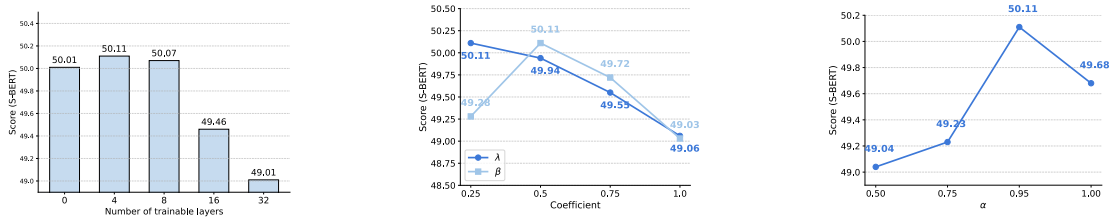

Figure A1. Fig. (Left) Impact of the number of LLM trainable layers during the stage 1 training. Fig. (Middle) Impact of the number of LLM trainable layers during stage 1 training. Fig(Right) Impact of group normalization coefficient on performance of SAGE-7B.

tioning task to demonstrate the stability of both the model and the training pipeline. Finally, we include a training-time analysis comparing SAGE with existing approaches in the literature.

A3.1. Comparison to Existing Methods

In Table A2, we compare SAGE with additional existing SOTA methods on 3D VQA and captioning tasks using the ScanQA [1] and Scan2Cap [11] datasets, respectively. On both 3D VQA and captioning tasks, SAGE outperforms the existing SOTA methods on most metrics. While no existing method dominantly performs best across all metrics, SAGE surpasses all existing methods in 4 out of 6 metrics across the 2 datasets, despite using less data for training and being computationally more efficient than existing methods.

A3.2. Training Time Analysis

In this section, we compare the total training time of our proposed method with PointLLM, which uses the same amount of training data and a similar training setup (Table A3). Here, the total training time refers to the end-to-end duration of the entire training pipeline. Due to its efficient model design, SAGE* requires considerably less training time (18 hours compared to 26.1 hours for PointLLM). The preference optimization step introduces additional training time; however, the complete three-stage training of our method still takes only slightly longer than the two-stage training pipeline of PointLLM.

A3.3. Zero-shot Classification

In Table A4, we compare the zero-shot classification performance of SAGE with existing SOTA methods on the ModelNet40 dataset. Here, SAGE shows improved or comparable performance with existing SOTA methods.

A3.4. Performance Across Multiple Runs

To assess the stability of our method, we report the mean and standard deviation of SAGE and SAGE* (7B parameters) across three independent runs in Table A5. Both variants demonstrate consistent performance with low variance across all metrics.

Table A4. **Performance comparison on Zero-shot 3D classification** on ModelNet40 [65] and ScanObjectNN [60]. Here, Ensembled [40] reprsents pretraining with four datasets, ShapeNet [6], ABO [14] and 3D-FUTURE [20]. †: Uni3D employs a larger EVA-CLIP-E [52] teacher, while other methods employ OpenCLIP-bigG [30].

Ours uses the Objaverse dataset paired with Cap3D generated captions.

Method	ModelNet40			ScanObjectNN		
	Top1	Top3	Top5	Top1	Top3	Top5
2D Inference without 3D Training						
PointCLIP [73]	19.3	28.6	34.8	10.5	20.8	30.6
PointCLIPv2 [80]	63.6	77.9	85.0	42.2	63.3	74.5
Trained on ShapeNet						
RECON [47]	61.2	73.9	78.1	42.3	62.5	75.6
CLIP2Point [29]	49.5	71.3	81.2	25.5	44.6	59.4
ULIP [69]	60.4	79.0	84.4	51.5	71.1	80.2
OpenShape [40]	70.3	86.9	91.3	47.2	72.4	84.7
TAMM [75]	73.1	88.5	91.9	54.8	74.5	83.3
MixCon3D [22]	72.6	87.1	91.3	52.6	69.9	78.7
Trained on Ensembled						
ULIP-2 [70]	75.1	88.1	93.2	51.6	72.5	82.3
OpenShape [40]	84.4	96.5	98.0	52.2	79.7	88.7
MixCon3D [22]	86.8	**96.9**	**98.3**	58.6	80.3	89.2
Uni3D-B† [77]	86.3	96.5	97.9	63.8	82.7	90.2
Uni3D-L† [77]	86.3	96.8	**98.3**	58.2	81.8	89.4
RECON++-B [48]	86.5	94.7	95.8	63.6	**84.2**	**90.6**
SAGE-7B	**88.9**	94.7	**98.3**	**65.8**	80.2	**90.6**

A4. Ablation and Sensitivity Studies

In this section, we perform an in-depth sensitivity analysis of various model-specific hyperparameters. We also conduct a detailed ablation study on the key components of our proposed model. All experiments are carried out using the 7B-parameter setting on the 3D captioning task with the Objaverse dataset, and we report the S-BERT alignment score.

A4.1. Sensitivity Study

Trainable layers. Our model, SAGE, utilizes LLaMA as the pretrained LLM backbone, which consists of 32 layers. In our proposed training pipeline, during the 3D tokenizer warm-up stage (stage 1), we train the first 4 layers of the LLM backbone. Since our proposed method doesn't

Table A5. Performance comparison across *three* runs on *(SAGE)* and *(SAGE*)* on 7B parameters. Here, the Objaverse dataset is used for 3D object captioning and recognition tasks, and the MM-Vet dataset is used for 3D VQA.

Model	Captioning						Cls.	VQA
	GPT-4	Sentence-BERT	SimCSE	BLEU-1	ROUGE-L	METEOR	GPT-4	GPT-4
SAGE-7B*	49.05	49.23	48.56	7.41	10.25	14.35	55.71	46.38
SD	0.09	0.04	0.05	0.04	0.02	0.02	0.03	0.04
SAGE-7B	50.98	50.11	49.70	9.50	12.66	16.95	57.11	49.53
SD	0.08	0.03	0.04	0.05	0.04	0.03	0.03	0.05

Table A6. Sensitivity study on different model-specific parameters on ModelNet40.

(a) Codebook size.

Size	S-BERT ↑
4096	48.88
8192	50.11
16384	49.76

(b) Number of point tokens.

Size	S-BERT ↑
128	48.98
512	50.11
1024	50.09

(c) Types of pooling layers.

Size	S-BERT ↑
Max pool	50.11
Avg. pool	49.89
Attn. pool	48.03

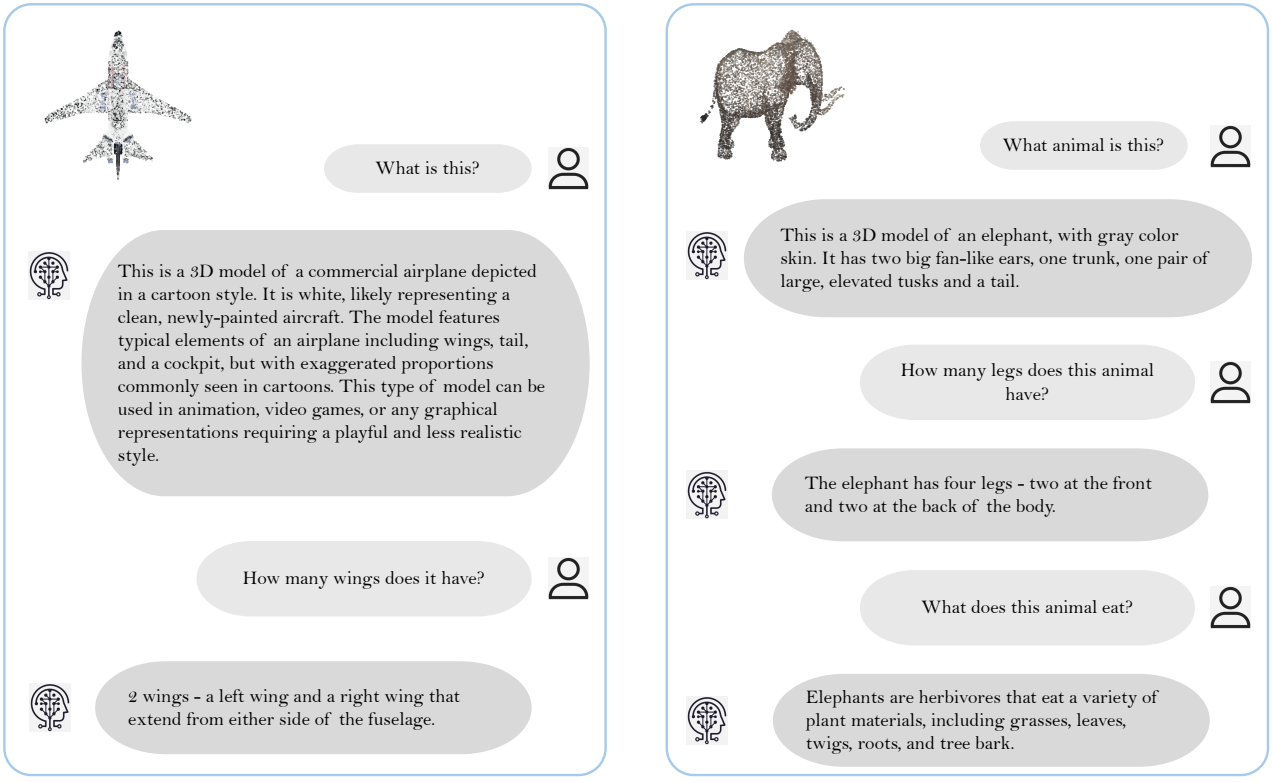

Figure A2. Dialogues between PointLLM and a human user.

Table A7. Impact of discrete vs. continuous point embeddings.

Embedding type	S-BERT ↑
Continuous (w/o codebook)	47.67
Discrete (with codebook)	50.11

utilize any pretrained 3D encoder, here the LLM solely extracts rich features from the point tokens along with the text tokens. In this experiment, we evaluate several configurations of the LLM backbone, ranging from fully freezing the

Table A8. Impact of different LLM backbones.

LLM	Model	S-BERT ↑
LLaMA-3.1	PointLLM	51.23
	SAGE	**55.89**
Qwen-2.5	PointLLM	52.35
	SAGE	**56.91**

LLM to partially tuning 4, 8, or 16 layers, as well as full fine-tuning of all 32 layers. From Figure A1*(left)*, we ob-

Table A9. Qualitative results on the Objaverse dataset.

Samples 1, 2		
Uid	059a7936ed89419ba9eae3153753ae86	82413f74efb84bc3b872a7a8f26cca81
Prompt	Caption this 3D model in detail.	Caption this 3D model in detail.
Human	A blue cartoon	A white fish with orange/red spots
InstructBLIP [15]	There is a cute blue and white pony in a still, looking cheerful and cheerful. It's standing next to a larger blue-colored object, potentially a barrier or structure, which provides the backdrop for this scene. The overall image is designed to create a fun, whimsical setting for the pony.	There is a fish hanging on a string in front of a black background. The fish is swimming in the middle of the image and is a little bit up from the bottom of the screen. The fish has a red spots on the left and right sides, and is positioned slightly towards the top of the screen.
LLaVA [39]	The image features a 3D model of a cute, colorful cartoon unicorn with a blue and white body and a hint of pink. The unicorn has a magical aura and is set against a black background. Its long, flowing mane and tail add to its whimsical charm. The unicorn appears to be in motion, possibly flying or prancing, as it captures the viewer's imagination.	The image features a 3D model of a fish floating in a dark, empty space. The fish is positioned towards the left side of the image and appears to be swimming upwards. The fish is large and occupies a significant portion of the image, giving the impression of a life-like, computer-generated representation. The dark background and lack of other objects in the scene emphasize the fish's presence and create a sense of depth and dimension.
3D-LLM [25]	A 3D model of a purple and pink pony with long hair, standing on its hind legs.	A small fish with a long tail is visible in the image. It has no distinguishing features or markings, and its color appears to be brown.
PointLLM [68]	A 3D model of a vibrant blue cartoon pony, characterized by its large white eyes, two wings, and a distinctive horn. The creature's overall design exudes a playful and friendly aura, making it appropriate for children's animations or toy production. The noticeably large white eyes and the unique wing and horn features endow the pony with an element of fantasy, while its blue body color gives it an overall appealing look.	The 3D model is of a long fish, illustrating a striking color combination of red, yellow, and grey. It's characterized by a slender tail. The yellow color is mostly seen on the body of the fish, fading into a subtler, grayish hue towards the fins and tail. The model depicts a typical long fish species, often found in the ocean, and can be used in a variety of digital applications, like animation, games, or virtual reality.
SAGE	**The 3D model is a depiction of a light blue pony, characterized by its vibrant color, large white eyes, and a notably long tail. It also has two vibrantly colored wings and a horn, which makes it look like a cartoon avatar.**	**The 3D model represents a fish with a streamlined body with a vibrant mix of color of yellow, red and silver. It has a long dorsal fin, two pectoral fins, a long anal fin, and large, captivating eyes.**

serve that the model's performance improves as 4 layers are trained compared to fully frozen LLM. Beyond this point, increasing the number of trainable layers leads to a performance drop, due to overfitting, given the relatively small size of the training dataset.

Model hyperparameters. In the vector quantization loss, β is a crucial hyperparameter that controls the contribution of the codebook loss and the quantization loss. Another important hyperparameter is λ, which balances the two components of the total training objective: the next-token prediction loss and the vector quantization loss. We investigate the impact of these two coefficients on SAGE's

performance through a sensitivity study. As shown in Figure A1 (middle), SAGE-7B achieves the highest performance gain with a quantization coefficient of 0.5 and a regularization coefficient of 0.25. Moreover, we observe that the performance is not highly sensitive to the choice of these hyperparameters.

Similarly, we examine the effect of the coefficient α in the reward combination during the policy optimization stage (stage 3). As shown in Figure A1 (right), the performance of SAGE-7B improves as α increases, reaching its highest point at $\alpha = 0.95$, which indicates that performance benefits from a lower weight on the length reward. However, performance drops when this reward is completely removed ($\alpha = 1.0$), highlighting the importance of using both rewards.

Codebook size. During vector quantization, the size of the learnable codebook defines the number of unique tokens allocated to the point cloud representations. Table A6a presents a sensitivity study on the codebook size used in the model. We vary the codebook size across three settings: 4096, 8192, and 16384. We observe that increasing the codebook size from 4096 to 8192 improves the score by 1.23. However, further increasing it to 16384 leads to a performance drop, indicating that around 8k+ tokens can sufficiently represent the point cloud features when treating it as a foreign language to LLM.

Number of point tokens. The number of point tokens defines the number of discrete geometric units the model uses to represent a 3D input, directly controlling the level of detail and the computational cost in an MLLM. Table A6b presents a sensitivity study on the number of point tokens used in the model. We vary this number across three settings: 128, 512, and 1024. Increasing the number of point tokens from 128 to 512 improves the score from 48.98 to 50.11. However, further increments do not lead to an improvement, even though they increase the computation. This suggests that 512-point tokens strike the optimal balance, maximizing the model's alignment capability. Therefore, we stick to 512 tokens for our final model.

Types of pooling functions. In table A6c, we investigate different pooling strategies for aggregating token representations in the proposed model. We evaluate three variants: max pooling, average pooling, and attention-based pooling. Here, using the max pooling gives the highest performance of 50.11, while average and attention pooling yield 49.89 and 49.03, respectively. These results indicate that max pooling is the most effective aggregation mechanism, due to its ability to preserve salient feature signals critical for cross-modal alignment.

A4.2. Ablation Study

Comparison of discrete vs. continuous embeddings. As discussed earlier, vector quantization bridges the gap between continuous geometric features and discrete language tokens by using a learnable codebook that maps continuous embeddings into a finite vocabulary of 3D tokens, effectively extending the LLM tokenizer to the 3D domain. To further validate its importance, we conduct an ablation study by comparing performance with and without the codebook in the 3D tokenizer. In Table A7, we can observe that with the codebook—using discrete embeddings—SAGE achieves a 50.11 score. Removing the codebook and keeping the embeddings continuous leads to a performance drop of 2.44. This confirms the necessity of vector quantization for enabling the LLM to effectively learn from multimodal signals.

A4.3. Generalization Across LLM Backbones

Finally, we investigate the generalization of SAGE to newer LLM backbones, namely LLaMA-3.1-8B and Qwen-2.5-7B. We also reproduce the results of PointLLM using these backbones. As shown in Table A8, SAGE demonstrates strong performance on both newer LLMs without any additional parameter tuning, and it consistently outperforms PointLLM.

A5. More Qualitative Results

We present more qualitative results on Objaverse datasets in Table A9. We further show interactive dialogues between a user and SAGE-7B in Figure A2, highlighting the model's strong understanding of point-cloud geometry, appearance, and functional attributes. The examples further demonstrate SAGE's ability to respond to user instructions with appropriate reasoning while avoiding biased or ill-informed outputs.

References

[1] Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19129–19139, 2022. 2, 3

[2] Jian Bai et al. Qwen: A family of large language models. *arXiv preprint arXiv:2309.16609*, 2023. 1

[3] Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, and Sagnak Tasırlar. Introducing our multimodal models, 2023. *URL https://www. adept. ai/blog/fuyu-8b*, 2, 2023. 3

[4] Daigang Cai, Lichen Zhao, Jing Zhang, Lu Sheng, and Dong Xu. 3djcg: A unified framework for joint dense captioning and visual grounding on 3d point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16464–16473, 2022. 2

[5] Tianyu Cai et al. Phi-3: Democratizing language model alignment. *arXiv preprint arXiv:2404.12345*, 2024. 1

[6] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2, 3

[7] Dave Zhenyu Chen, Qirui Wu, Matthias Nießner, and Angel X Chang. D3net: A speaker-listener architecture for semi-supervised dense captioning and visual grounding in rgb-d scans. 2021. 2

[8] Sijin Chen, Hongyuan Zhu, Xin Chen, Yinjie Lei, Gang Yu, and Tao Chen. End-to-end 3d dense captioning with vote2cap-detr. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11124–11133, 2023. 2

[9] Yangyi Chen, Xingyao Wang, Hao Peng, and Heng Ji. Solo: A single transformer for scalable vision-language modeling. *Transactions on Machine Learning Research*, 2024. 3

[10] Yilun Chen, Shuai Yang, Haifeng Huang, Tai Wang, Runsen Xu, Ruiyuan Lyu, Dahua Lin, and Jiangmiao Pang. Grounded 3d-llm with referent tokens. *arXiv preprint arXiv:2405.10370*, 2024. 2

[11] Zhenyu Chen, Ali Gholami, Matthias Nießner, and Angel X Chang. Scan2cap: Context-aware dense captioning in rgb-d scans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3193–3203, 2021. 2, 3

[12] Zhenyu Chen, Ronghang Hu, Xinlei Chen, Matthias Nießner, and Angel X Chang. Unit3d: A unified transformer for 3d dense captioning and visual grounding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 18109–18119, 2023. 2

[13] Wei-Lin Chiang, Zhuohan Li, Ziqing Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2(3):6, 2023. 5

[14] Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F Yago Vicente, Thomas Dideriksen, Himanshu Arora, et al. Abo: Dataset and benchmarks for real-world 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21126–21136, 2022. 2, 3

[15] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in neural information processing systems*, 36:49250–49267, 2023. 7, 8, 5

[16] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13142–13153, 2023. 6, 2

[17] Jiajun Deng, Tianyu He, Li Jiang, Tianyu Wang, Feras Dayoub, and Ian Reid. 3d-llava: Towards generalist 3d lmms with omni superpoint transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3772–3782, 2025. 3, 2

[18] Haiwen Diao, Yufeng Cui, Xiaotong Li, Yueze Wang, Huchuan Lu, and Xinlong Wang. Unveiling encoder-free vision-language models. *Advances in Neural Information Processing Systems*, 37:52545–52567, 2024. 3

[19] Haiwen Diao, Xiaotong Li, Yufeng Cui, Yueze Wang, Haoge Deng, Ting Pan, Wenxuan Wang, Huchuan Lu, and Xinlong Wang. Evev2: Improved baselines for encoder-free vision-language models. *arXiv preprint arXiv:2502.06788*, 2025. 3

[20] Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3d-future: 3d furniture shape with texture. *International Journal of Computer Vision*, 129(12):3313–3337, 2021. 2, 3

[21] Rao Fu, Jingyu Liu, Xilun Chen, Yixin Nie, and Wenhan Xiong. Scene-llm: Extending language model for 3d visual understanding and reasoning. *arXiv preprint arXiv:2403.11401*, 2024. 2

[22] Yipeng Gao, Zeyu Wang, Wei-Shi Zheng, Cihang Xie, and Yuyin Zhou. Sculpting holistic 3d representation in contrastive language-image-3d pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22998–23008, 2024. 2, 3

[23] Yijie Guo et al. Ll3d: Large language models meet 3d. *arXiv preprint arXiv:2312.04837*, 2023. 2

[24] Ziyu Guo, Renrui Zhang, Xiangyang Zhu, Yiwen Tang, Xianzheng Ma, Jiaming Han, Kexin Chen, Peng Gao, Xianzhi Li, Hongsheng Li, et al. Point-bind & point-llm: Aligning point cloud with multi-modality for 3d understanding, generation, and instruction following. *arXiv preprint arXiv:2309.00615*, 2023. 2, 3

[25] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *Advances*

in Neural Information Processing Systems, 36:20482–20494, 2023. 8, 2, 5

[26] Wenbo Hu, Yining Hong, Yanjun Wang, Leison Gao, Zibu Wei, Xingcheng Yao, Nanyun Peng, Yonatan Bitton, Idan Szpektor, and Kai-Wei Chang. 3dllm-mem: Long-term spatial-temporal memory for embodied 3d large language model. *arXiv preprint arXiv:2505.22657*, 2025. 3

[27] Haifeng Huang, Yilun Chen, Zehan Wang, Rongjie Huang, Runsen Xu, Tai Wang, Luping Liu, Xize Cheng, Yang Zhao, Jiangmiao Pang, et al. Chat-scene: Bridging 3d scene and large language models with object identifiers. *Advances in Neural Information Processing Systems*, 37:113991–114017, 2024. 2

[28] Jiangyong Huang, Silong Yong, Xiaojian Ma, Xiongkun Linghu, Puhao Li, Yan Wang, Qing Li, Song-Chun Zhu, Baoxiong Jia, and Siyuan Huang. An embodied generalist agent in 3d world. *arXiv preprint arXiv:2311.12871*, 2023. 2

[29] Tianyu Huang, Bowen Dong, Yunhan Yang, Xiaoshui Huang, Rynson WH Lau, Wanli Ouyang, and Wangmeng Zuo. Clip2point: Transfer clip to point cloud classification with image-depth pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22157–22167, 2023. 3

[30] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, et al. Openclip, july 2021. *URL https://doi. org/10.5281/zenodo*, 5143773(2):29, 2021. 2, 3

[31] Albert Q. Jiang et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023. 1

[32] Yang Jiao, Shaoxiang Chen, Zequn Jie, Jingjing Chen, Lin Ma, and Yu-Gang Jiang. More: Multi-order relation mining for dense captioning in 3d scenes. In *European Conference on Computer Vision*, pages 528–545. Springer, 2022. 2

[33] Zhao Jin, Munawar Hayat, Yuwei Yang, Yulan Guo, and Yinjie Lei. Context-aware alignment and mutual masking for 3d-language pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10984–10994, 2023. 2

[34] Weitai Kang, Haifeng Huang, Yuzhang Shang, Mubarak Shah, and Yan Yan. Robin3d: Improving 3d large language model via robust instruction tuning. pages 3905–3915, 2025. 3, 2

[35] Bo Li, Peiyuan Zhang, Jingkang Yang, Yuanhan Zhang, Fanyi Pu, and Ziwei Liu. Otterhd: A high-resolution multi-modality model. *arXiv preprint arXiv:2311.04219*, 2023. 3

[36] Rong Li, Shijie Li, Lingdong Kong, Xulei Yang, and Junwei Liang. Seeground: See and ground for zero-shot open-vocabulary 3d visual grounding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3707–3717, 2025. 3

[37] Xiang Li et al. Gpt-4v(ision): Multimodal reasoning across text and images. *arXiv preprint arXiv:2402.00123*, 2024. 1

[38] Haotian Liu et al. Llava-next: Stronger vision-language understanding with large multimodal models. *arXiv preprint arXiv:2401.03124*, 2024. 1

[39] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 7, 8, 5

[40] Minghua Liu, Ruoxi Shi, Kaiming Kuang, Yinhao Zhu, Xuanlin Li, Shizhong Han, Hong Cai, Fatih Porikli, and Hao Su. Openshape: Scaling up 3d shape representation towards open-world understanding. *Advances in neural information processing systems*, 36:44860–44879, 2023. 2, 3

[41] Tiange Luo, Chris Rockwell, Honglak Lee, and Justin Johnson. Scalable 3d captioning with pretrained models. *Advances in Neural Information Processing Systems*, 36:75307–75337, 2023. 6

[42] Chenyang Ma, Kai Lu, Ta-Ying Cheng, Niki Trigoni, and Andrew Markham. Spatialpin: Enhancing spatial reasoning capabilities of vision-language models through prompting and interacting 3d priors. *Advances in neural information processing systems*, 37:68803–68832, 2024. 3

[43] Yunze Man, Liang-Yan Gui, and Yu-Xiong Wang. Situational awareness matters in 3d vision language reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13678–13688, 2024. 3

[44] Youquan Pang et al. Masked autoencoders for point cloud self-supervised learning. *ECCV*, 2022. 1

[45] Yatian Pang, Eng Hock Francis Tay, Li Yuan, and Zhenghua Chen. Masked autoencoders for 3d point cloud self-supervised learning. *World Scientific Annual Review of Artificial Intelligence*, 1:2440001, 2023. 3

[46] Hengshuang Qi et al. Contrastive 3d learning with multi-level alignment. *NeurIPS*, 2023. 1

[47] Zekun Qi, Runpei Dong, Guofan Fan, Zheng Ge, Xiangyu Zhang, Kaisheng Ma, and Li Yi. Contrast with reconstruct: Contrastive 3d representation learning guided by generative pretraining. In *International Conference on Machine Learning*, pages 28223–28243. PMLR, 2023. 2, 3

[48] Zekun Qi, Runpei Dong, Shaochen Zhang, Haoran Geng, Chunrui Han, Zheng Ge, Li Yi, and Kaisheng Ma. Shapellm: Universal 3d object understanding for embodied interaction. In *European Conference on Computer Vision*, pages 214–238. Springer, 2024. 2, 4, 6, 7, 3

[49] Zhangyang Qi, Zhixiong Zhang, Ye Fang, Jiaqi Wang, and Hengshuang Zhao. Gpt4scene: Understand 3d scenes from videos with vision-language models. *arXiv preprint arXiv:2501.01428*, 2025. 3

[50] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 5

[51] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. 5

[52] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023. 2, 3

[53] Yiwen Tang, Zoey Guo, Zhuhao Wang, Ray Zhang, Qizhi Chen, Junli Liu, Delin Qu, Zhigang Wang, Dong Wang, Xuelong Li, et al. Exploring the potential of encoder-free archi-

tectures in 3d lmms. *arXiv preprint arXiv:2502.09620*, 2025. 3

[54] Yuan Tang, Xu Han, Xianzhi Li, Qiao Yu, Yixue Hao, Long Hu, and Min Chen. Minigpt-3d: Efficiently aligning 3d point clouds with large language models using 2d priors. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 6617–6626, 2024. 3

[55] Yuan Tang, Xu Han, Xianzhi Li, Qiao Yu, Jinfeng Xu, Yixue Hao, Long Hu, and Min Chen. More text, less point: Towards 3d data-efficient point-language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 7284–7292, 2025. 3

[56] Zhiqi Tang et al. Geometryclip: Bridging 3d geometry and language through multimodal pretraining. *arXiv preprint arXiv:2402.06712*, 2024. 1

[57] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024. 3

[58] Anh Thai, Songyou Peng, Kyle Genova, Leonidas Guibas, and Thomas Funkhouser. Splattalk: 3d vqa with gaussian splatting. *arXiv preprint arXiv:2503.06271*, 2025. 3

[59] Hugo Touvron et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 1, 5

[60] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1588–1597, 2019. 2, 3

[61] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 4

[62] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3

[63] Heng Wang, Chaoyi Zhang, Jianhui Yu, and Weidong Cai. Spatiality-guided transformer for 3d dense captioning on point clouds. *arXiv preprint arXiv:2204.10688*, 2022. 2

[64] Haochen Wang, Yucheng Zhao, Tiancai Wang, Haoqiang Fan, Xiangyu Zhang, and Zhaoxiang Zhang. Ross3d: Reconstructive visual instruction tuning with 3d-awareness. *arXiv preprint arXiv:2504.01901*, 2025. 3

[65] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015. 2, 3

[66] Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *The Thirteenth International Conference on Learning Representations*, 2025. 3

[67] Saining Xie et al. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. *ECCV*, 2020. 1, 3

[68] Tianyu Xu et al. Pointllm: Empowering large language models to understand point clouds. *CVPR*, 2025. 1, 3, 4, 5, 6, 7, 8, 2

[69] Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. Ulip: Learning a unified representation of language, images, and point clouds for 3d understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1179–1189, 2023. 2, 3

[70] Le Xue, Ning Yu, Shu Zhang, Artemis Panagopoulou, Junnan Li, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, et al. Ulip-2: Towards scalable multimodal pre-training for 3d understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27091–27101, 2024. 2, 3

[71] Hanxun Yu, Wentong Li, Song Wang, Junbo Chen, and Jianke Zhu. Inst3d-lmm: Instance-aware 3d scene understanding with multi-modal instruction tuning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 14147–14157, 2025. 3

[72] Xumin Yu et al. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. *CVPR*, 2022. 1, 3

[73] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. Pointclip: Point cloud understanding by clip. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8552–8562, 2022. 2, 3

[74] Yichi Zhang et al. I2p-mae: Image-to-point masked autoencoders for self-supervised learning on 3d data. *ICCV*, 2023. 1

[75] Zhihao Zhang, Shengcao Cao, and Yu-Xiong Wang. Tamm: Triadapter multi-modal learning for 3d shape understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21413–21423, 2024. 2, 3

[76] Duo Zheng, Shijia Huang, and Liwei Wang. Video-3d llm: Learning position-aware video representation for 3d scene understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8995–9006, 2025. 3

[77] Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3d: Exploring unified 3d representation at scale. *arXiv preprint arXiv:2310.06773*, 2023. 2, 3

[78] Chenming Zhu, Tai Wang, Wenwei Zhang, Jiangmiao Pang, and Xihui Liu. Llava-3d: A simple yet effective pathway to empowering lmms with 3d-awareness. *arXiv preprint arXiv:2409.18125*, 2024. 3, 2

[79] Jian Zhu, Hanli Wang, and Miaojing Shi. Multi-modal large language model enhanced pseudo 3d perception framework for visual commonsense reasoning. *IEEE Transactions on Circuits and Systems for Video Technology*, 2024. 3

[80] Xiangyang Zhu, Renrui Zhang, Bowei He, Ziyu Guo, Ziyao Zeng, Zipeng Qin, Shanghang Zhang, and Peng Gao. Pointclip v2: Prompting clip and gpt for powerful 3d open-world learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2639–2650, 2023. 2, 3

[81] Ziyu Zhu, Xiaojian Ma, Yixin Chen, Zhidong Deng, Siyuan Huang, and Qing Li. 3d-vista: Pre-trained transformer for 3d

vision and text alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2911–2921, 2023. 2